Calculations Behind Lottery Valuations*

Hanyao Zhang†

August 8, 2025

Click here for the most recent version

Abstract

I introduce a novel experimental design tracking subjects' calculations when valuing lotteries. Subjects' calculations predominantly fall into three groups: expected values, linear functions of potential monetary outcomes, or expressions that cannot be matched to primitives of the lotteries. Across different tasks, the calculations exhibit remarkable within-subject stability alongside substantial between-subject heterogeneity. Calculations strongly predict valuations: subjects performing calculations related to the expected values (38.1%) exhibit near risk-neutrality, while others' (61.9%) valuations on average display extreme unresponsiveness to probability changes. Finally, an analysis by calculation group reveals that distinct theoretical mechanisms drive these behaviors: the adoption of expected-value calculations is explained by a reduction in implementation costs from the provided calculator, while attribute substitution (Kahneman and Frederick, 2002) explains the linear functions of potential monetary outcomes.

*I am indebted to Mark Dean for his invaluable guidance and unwavering support. I also benefited from comments by Hassan Afrouzi, Alessandra Casella, Navin Kartik, Judd Kessler, Zhi Hao Lim, Kirby Nielsen, Ryan Oprea, Charlie Sprenger, Mike Woodford, Songfa Zhong, Yangfan Zhou, members of the Columbia Cognition and Decision Lab, seminar audience at Columbia, BRIC XI (ITAM), Caltech CTESS Workshop, SDM III (SWUFE), and SWEET (UPenn). Hao Lin and Chris Yuan provided excellent research assistance. I am grateful for the financial support from Columbia CELSS. This research was approved by Columbia IRB. All mistakes are my own.

†Department of Economics, Columbia University, New York. Email: hanyao.zhang@columbia.edu.

1 Introduction

A common finding in the literature on risk attitudes is that economic agents' valuations over risky lotteries are unresponsive to changes in probabilities: When probabilities of the potential monetary outcomes change, the resulting changes in lottery *valuations* are generally smaller in magnitude than the changes in lottery *expected values*.[1] The pattern of unresponsiveness unifies the fourfold pattern documented in Tversky and Kahneman (1992):[2] risk aversion for gains of high probability and losses of low probability; risk seeking for gains of low probability and losses of high probability. The unresponsiveness is also a leading example of the broader theme of behavioral attenuation that has been observed across multiple decision-making domains (Enke et al., 2024).

The literature studying unresponsiveness has mainly focused on "as-if" models that capture this phenomenon through descriptive parameters but do not aim to describe the actual decision-making processes underlying the valuations.[3] For example, commonly interpreted as an "as-if" model, prospect theory accounts for unresponsiveness through the relatively flat region in the mid-range probabilities of the inverse-S-shaped probability weighting function (e.g, Tversky and Kahneman, 1992, Gonzalez and Wu, 1999). While such models successfully describe observed choice patterns, understanding the underlying decision-making processes that lead to unresponsive lottery valuations represents a crucial next step. The decision-making processes shed light on the *cause* of unresponsiveness, which goes beyond the descriptive *fit* of the "as-if" models, and have important implications for both theory and applications.

In this study, I focus on a specific aspect of the decision-making processes: the calculations performed when valuing lotteries. To identify these calculations, I conduct an online experiment in which I provide subjects with a calculator on the experimental interface when they value lotteries, and track the calculations the subjects perform with the calculator. These calculations provide a unique window into the processes by which lottery valuations are

[1] Strictly speaking, unresponsiveness may disappear under extreme probability values, for example if the probability changes from 0.99 to 0.999. However, this paper restricts attention to probability values that are bounded away from zero and one (between 8% and 92%).

[2] To see how unresponsiveness unifies the fourfold pattern, see Section 3, and also Blavatskyy (2007).

[3] Notable exceptions include Payne, Bettman and Johnson (1988), Arieli, Ben-Ami and Rubinstein (2011), Pachur et al. (2013, 2018), and Arrieta and Nielsen (2023).

generated – both the deliberately-chosen computational procedures and the more automatic heuristics. Further, the data allow me to classify subjects by the calculations they use, and ultimately, to compare the observed calculations with the predictions of behavioral theories of decisions under risk.

The lotteries included in this study are binary-outcome lotteries that pay \$26 or lose \$26 with probability $p \in \{8\%, 25\%, 75\%, 92\%\}$, and \$0 otherwise. In addition to the main treatment that implements the calculator design (*Calc*), I also implement another within-subject treatment that drops the calculator from the experimental interface (*NoCalc*). Moreover, using the same set of subjects, I elicit valuations of the deterministic mirrors of each lottery (Oprea, 2024*b*), under both NoCalc and Calc treatments. For each lottery, its deterministic mirror is presented in a similar, disaggregated form as the lottery – a sequence of monetary outcomes and their weights – but pays out the corresponding lottery's expected value *with certainty.* The mirrors preserve some aspects of the lotteries (the need to integrate various outcomes and probabilities together when making a decision) while removing risk and its resulting unknown risk preferences. I study mirrors to understand how subjects make decisions under a problem with an unambiguous correct answer that is closely related to the lottery tasks, with the goal of examining whether subjects' calculations and mistakes in the mirror tasks are related to how they approach the lottery tasks.

The lottery valuations from the Calc treatment reproduce the well-documented unresponsiveness: when the probability of being paid or suffering a loss changes, the resulting changes in average valuation are smaller than those of the expected value. Comparing the Calc and NoCalc treatments, the average subject is more responsive in the Calc treatment, but only by a small magnitude. Echoing Oprea (2024*b*), unresponsiveness also appears in the valuations of deterministic mirrors.

Next, I analyze the calculations performed by subjects. My data consist of sequences of numerical expressions that the subjects calculate. This novel data present several key challenges. First, the calculation data only captures the part of subjects' decision-making processes that is based on *explicit calculation rules*, as opposed to process-opaque decision-making. Second, even when the decision-making processes are based on explicit calculation rules, for them to be captured in the data, the subjects have to actually perform the explicit calculation rules in the calculator, not in their minds. I refer to these two challenges as the *incomplete record* limitation of the data. Given this limitation, my analysis focuses on extracting meaningful insights from the calculations that subjects do choose to perform explicitly. While I cannot recover complete decision-making processes when subjects rely primarily on process-opaque decision-making or mental calculations, the data still provide valuable information about the explicit computational strategies subjects employ when they do engage with the calculator. Finally, the calculator inputs are high-dimensional objects requiring transformation into summary features for quantitative analysis, and the features constructed inevitably capture some aspects of the calculations while missing others.

Using the calculator input data from the Calc treatment, I first address the descriptive question: what do subjects calculate when they determine their lottery valuations? Subjects predominantly employ one of three main calculation groups (ordered by frequency): (1) calculations related to expected values (EV); (2) calculations that cannot be matched with any task primitives (monetary outcomes and probabilities); (3) calculations that are linear functions of only the monetary outcomes. Critically, individual subjects typically commit to one calculation group rather than mixing calculation groups within a valuation task.

Moreover, the calculations exhibit within-subject stability across tasks – a fixed subject generally uses similar calculations across different tasks. This within-subject stability extends beyond lottery tasks: for a majority of subjects, the calculations employed for lottery tasks are similar to those employed for mirror tasks. This observation suggests that their underlying decision-making processes for valuing lotteries and mirrors share some similarity.

Given the within-subject stability of calculations, I categorize subjects based on their calculations in lottery tasks. This yields five subject types: three primary types that directly correspond to the three main calculation groups – (1) the EV type, (2) the number type, and (3) the linear money type – plus two minor types. The analysis proceeds in two steps. First, I examine how subject types relate to valuations across both treatments, and in both lottery and mirror tasks. Second, for each type, I identify theoretical mechanisms that explain the joint patterns of calculations and valuations. I present this analysis by examining each type in turn.

EV Type The EV-type subjects (38.1% of all subjects), characterized by their use of calculations related to the expected values, tend to submit highly responsive lottery valuations and appear close to risk neutral in the Calc treatment. To quantify this responsiveness, I estimate separate regressions for each subject using their lottery valuation data, where I regress their valuations of the lotteries on the expected values of these lotteries. I then use each subject's estimated slope on expected value as an individual-level measure of responsiveness. The analysis reveals that the average responsiveness of the EV-type subjects is 0.85. In other words, when the expected value of the lottery increases by $1, on average the valuation of the lottery increases by $0.85.

In the *mirror* tasks of the Calc treatment, the EV-type subjects are also highly responsive (average = 0.82). In the lottery tasks of the *NoCalc* treatment, the EV-type subjects are much less responsive than in the Calc treatment, but are still moderately responsive (average = 0.56). Moreover, when given an incentivized opportunity to reconcile their inconsistent choices across the two treatments (the *reconciliation stage,* Nielsen and Rehbeck, 2022), these EV-type subjects generally prefer their responsive valuations in the Calc treatment over their unresponsive valuations in the NoCalc treatment, which suggests that EV-type subjects consider their responsive, near-risk-neutral valuations in the Calc treatment to more accurately reflect their welfare-relevant risk preferences than their moderately responsive valuations in the NoCalc treatment.

Next, I examine the theoretical mechanisms that can explain their unresponsiveness in the NoCalc treatment, which resembles experiments in the previous literature that measure risk attitudes. Their dramatic increase in responsiveness from the NoCalc to the Calc treatment is consistent with the theory of implementation costs – costs of implementing procedures despite knowledge of optimal approaches. Although these subjects exhibit near-risk-neutral preferences, as evident by their choices in the reconciliation stage, implementing the computational procedure that optimizes against this preference – namely, calculating expected values – involves implementation costs. When the implementation costs are relatively high due to the absence of a calculator in the NoCalc treatment, these subjects resort to less costly decision-making processes that generate unresponsive valuations. In contrast, the Calc treatment lowers the implementation costs of their optimal procedure due to the presence of a calculator, and as a result, these EV-type subjects choose to implement their optimal procedure, and exhibit high responsiveness and near risk-neutrality.

Number Type The number type subjects (36.6% of all subjects), characterized by their use of calculations that cannot be matched with any task primitives, tend to submit highly unresponsive lottery valuations (average = 0.20) and appear well-described by the fourfold pattern of risk attitudes (Tversky and Kahneman, 1992) in the Calc treatment. These subjects are also highly unresponsive in both the *mirror* tasks of the Calc treatment (average = 0.21) and the lottery tasks of the *NoCalc* treatment (average = 0.25). The valuations of the number type subjects are statistically indistinguishable between the NoCalc and Calc treatments.

The calculator design provides limited insights into why the number type subjects exhibit unresponsiveness, since their calculations do not reveal much about how the subjects map the task primitives to their valuations. However, what I do observe is that these subjects also largely use the same group of calculations – those that cannot be matched with any task primitives – in mirror tasks, when objectively correct answers are available that can be reached by simply calculating the expected values. Perhaps a bit speculatively, this suggests that the unresponsiveness of the number type in lottery tasks is not solely driven by the fact that lotteries are risky, but partly related to the fact that lotteries are disaggregated objects.

Linear Money Type Despite employing a different calculation approach – linear functions of monetary outcomes – linear money type subjects (16.8% of all subjects) also exhibit high unresponsiveness across all treatments and across lottery and mirror tasks (lottery Calc: 0.26, mirror Calc: 0.25, lottery NoCalc: 0.25), suggesting that their explicit calculations do not translate into more responsive decision-making.

I examine whether two specific theories – probability weighting and attribute substitution – can potentially explain the joint patterns of their calculations and valuations. I begin by evaluating probability weighting as a potential explanation, since if subjects are literally implementing probability weighting to determine valuations, their calculations can be shown to appear as a linear function of monetary outcomes in the data. To test this possibility, I recover the probability weighting function implied by calculations, contrasting with traditional approaches that infer probability weights from valuations. The recovered probability weighting function shows an extremely flat slope (0.093) with respect to the actual probability, indicating that these subjects apply similar weights across widely different probabilities. The extreme degree of unresponsiveness is difficult to reconcile with probability weighting as a literal account.

Instead, the emergence of linear money type subjects can be explained by the attribute substitution theory (Kahneman and Frederick, 2002), where subjects generate their valuations mainly based on the potential monetary outcomes of the lotteries. This is because their genuine valuations may be relatively inaccessible, while potential monetary outcomes are semantically related to the valuations (since they both involve monetary amounts) and are readily accessible, and thus can serve as a substitute for the valuations. The theory explains the observed unresponsiveness to probabilities through the neglect of probabilities in the attribute substitution process. This theory can also explain an important empirical pattern documented across the data from multiple studies but has not attracted sufficient attention: lottery valuations are substantially more responsive to changes in monetary outcomes than to changes in probabilities. Finally, the most prevalent form of attribute substitution directly observed in the calculation data generates a novel testable prediction: responsiveness to a monetary outcome should decrease as its probability increases. This prediction is confirmed with Enke and Graeber's (2023) data.

Summarizing the type-by-type analysis above, the calculation data reveal three primary subject types: the EV type, the number type, and the linear money type. Examining the theoretical mechanisms underlying unresponsiveness reveals distinct mechanisms by type. The unresponsiveness exhibited by EV-type subjects in the NoCalc treatment predominantly reflects implementation costs. In contrast, the calculations and unresponsive valuations of linear money-type subjects provide strong evidence of attribute substitution, where subjects substitute the readily accessible monetary outcomes for the more complex lottery valuation task, thereby neglecting probabilities. For number-type subjects, the underlying mechanism remains less clear. Among the remaining subjects in the minor types, I find evidence that many suffer from incomplete understanding of the lottery valuation task.

The rest of this paper is organized as follows. Section 2 describes my experimental design. Section 3 describes subjects' valuations. Section 4 outlines the methodology of analyzing the calculations, and provides descriptions of the calculations arising in my experimental data. Section 5 links the calculations to the valuations, and Section 6 discusses the implications of the calculations over the theoretical mechanisms underlying unresponsive lottery valuations. Finally, Section 7 discusses how the current study relates to the literature.

2 Experimental Design

2.1 Lotteries and Their Deterministic Mirrors

In the experiment, I elicit the valuations (certainty equivalents) for a set of 8 distinct lotteries using the Becker-Degroot-Marschak (BDM) mechanism (Becker, Degroot and Marschak, 1964). I focus on simple, two-outcome lotteries ($\$X, p; 0$) ($\$X$ is paid with probability p, and \$0 is paid with the remaining probability). Two groups of lotteries are included. In the first group, "gain lotteries," $X = 26$ and the subject gains \$26 with probabilities $p \in \{0.08, 0.25, 0.75, 0.92\}$. These lotteries are referred to as G8, G25, G75, and G92, respectively. In "loss lotteries," $X = -26$ and the subject loses \$26 with probabilities $p \in \{0.08, 0.25, 0.75, 0.92\}$. These lotteries are referred to as L8, L25, L75, and L92, respectively. To measure valuation inconsistency and its relationship with valuation patterns, the lottery G25 is repeated, leading to a total of 9 lottery *tasks*.

The experimental instructions depict the gain (loss) lottery Gn (Ln) as 100 boxes, of which n contain \$26 (-\$26) and the rest contain \$0. To determine the payment from the lottery, one of these boxes will be randomly selected, and the amount of money in the selected box will be paid. Following Oprea (2024*b*), I elicit the valuations of the *deterministic mirror* of each lottery, with the mirror of lottery G25 again repeated. A deterministic mirror is presented in a similar format as its corresponding lottery, but features a modified payoff rule that eliminates risk and pays the expected value of its corresponding lottery with certainty. Specifically, a mirror is also depicted as 100 boxes, each containing some amount of money. However, instead of paying out a randomly selected box like a lottery does, a mirror pays the average amount of money across the 100 boxes. I use tuples such as (G8, lottery) and (L25, mirror) to refer to individual valuation tasks, and use *task type* to refer to the two different payoff rules: Lottery and mirror.

Addressing the concerns raised by Banki et al. (2025) and **?** over Oprea's (2024*b*) instructions, which I adopt extensively, I replicate all the analysis in this paper with two exercises. First, I replicate the main experiment using an entirely new set of instructions that combine **?**'s (**?**) explanations of lotteries and mirrors, and Healy's (2020) explanations of the BDM mechanism. Second, I use the data from the main experiment, but restrict the sample to those subjects who perfectly answer all comprehension questions and are unlikely to be confused about the experimental design. The vast majority of the results in this paper are robust in both exercises. See Appendix D for a more thorough discussion.

2.2 Experimental Treatments

The experiment includes two within-subject treatments: *NoCalc* and *Calc*.

NoCalc Treatment In the NoCalc treatment, the subject values the 9 lotteries and mirrors by typing their valuations into a text box. The subject is given \$30 as their initial money for them to bid under the BDM mechanism, and their gains and losses are calculated on top of the initial money. The valuations are restricted to be between \$0 and \$26 for tasks involving gains, and between -\$26 and \$0 for tasks involving losses.

Calc Treatment In the Calc treatment, the subject values the same 9 lotteries and mirrors. The key innovation is the inclusion of a calculator in the experimental interface, whose input I can track and record. The calculator can perform basic arithmetic operations. Numbers and operations can be typed into the calculator by either clicking the buttons on the graphical interface or using a keyboard. The calculator can store multiple calculations. All expressions calculated are displayed in the calculator as a table, in the order of being performed. Each line in the table consists of a *Calculation* column, where the expression calculated is displayed, and a *Result* column, where the calculated result appears. The calculator refreshes after each task, clearing all previous expressions and results. A screenshot of the experimental interface with example calculations performed can be seen in Figure 1.

As in the NoCalc treatment, a text box is provided asking for the valuation of the subject (seen on the left of Figure 1). However, the text box is grayed out, and the subject is not able to directly type numbers into the text box. Instead, to submit a valuation, the subject needs to make it appear in the Result column of the last line of the calculator. Then, the

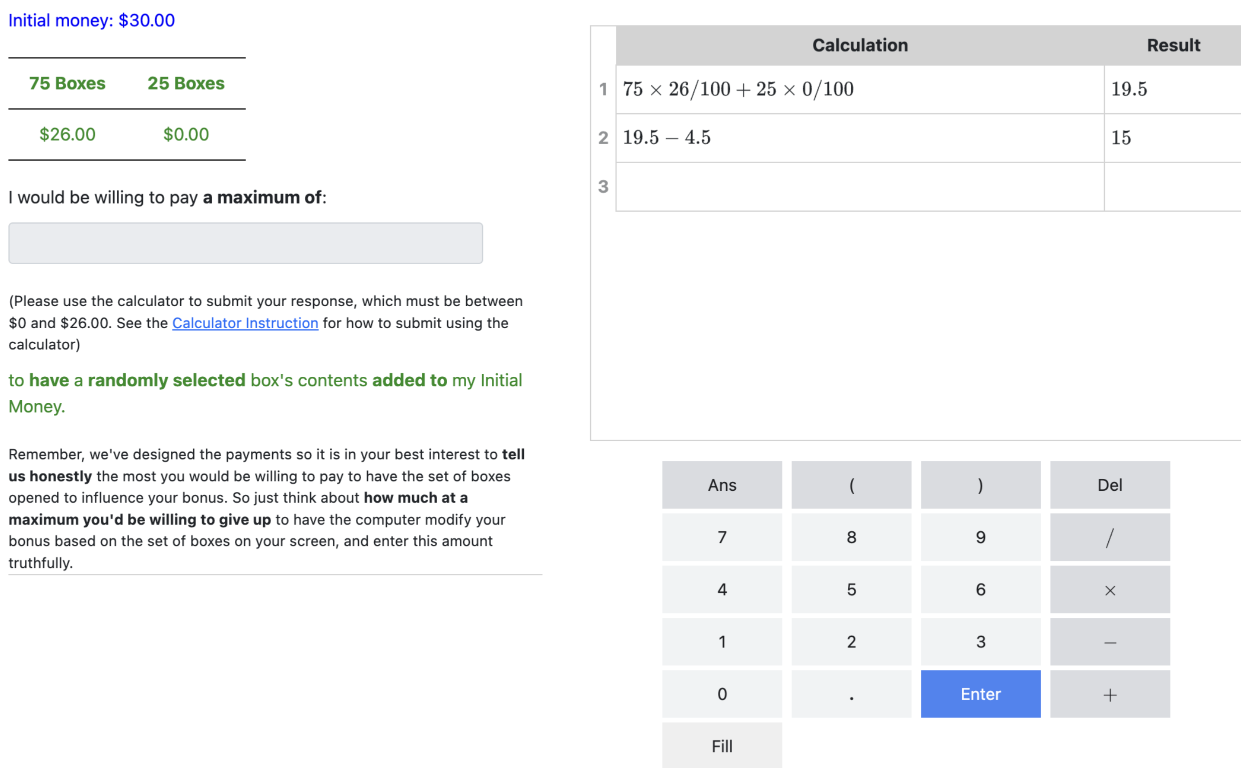


Figure 1: The experimental interface in the Calc treatment with example calculations performed

subject would click the *Fill* button (shown in the bottom-left of the calculator interface in Figure 1) to fill the number into the grayed-out text box. This submission process is designed to balance two goals. On the one hand, it gently encourages subjects to use the calculator. On the other hand, it minimizes potential distortions of behaviors. Particularly, the design accommodates subjects who wish to submit valuations without performing calculations in the calculator. Such a subject can simply type their intended valuation as a single number in the calculator, and then "calculate" this number, and finally fill the number into the text box.

The subject does not receive any specific instructions as to how the calculator may help them in the task, and they are free to use the calculator to perform whatever calculations they deem useful. The payment of the subject does not depend on what calculations the subject performs in the calculator, and only depends on the valuations that they submit.

Timeline The experiment starts with the NoCalc treatment. The NoCalc treatment consists of two blocks – one containing all lottery tasks and the other containing all mirror tasks. The order of blocks and the order of tasks within each block are randomized at the subject level. The subject is not informed about the second block while completing the first block, but receives instructions about the new task type before starting the second block.

After finishing the NoCalc treatment, the subject enters the Calc treatment. Within the Calc treatment, there are again two blocks for lottery tasks and mirror tasks, respectively. The order of lottery and mirror blocks in the Calc treatment is the same as that in the NoCalc treatment. Figure 2 shows the diagram for the main experimental timeline.

Before each of the four blocks, the subject is required to answer four comprehension questions. These comprehension questions serve the purpose of training the subjects on the payoff rules in lottery tasks and mirror tasks, and also as a reminder that the payoff rule has changed from the one used previously. The four comprehension questions are shown on the same screen. The subject has unlimited opportunities to answer the questions, but they have to answer all four questions correctly at a single trial in order to proceed.

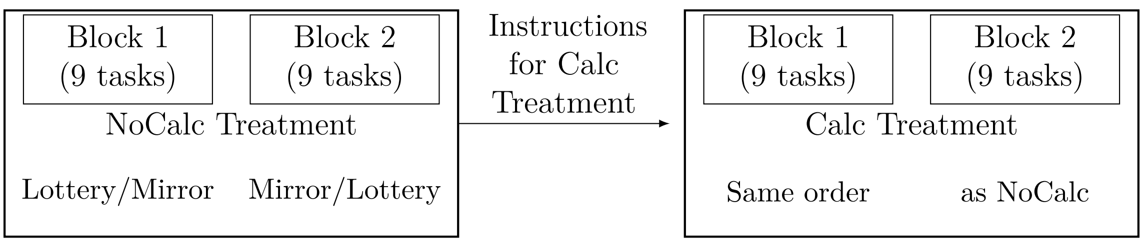


Figure 2: Main Experimental Timeline

After the Calc treatment, following Nielsen and Rehbeck (2022), the subject is asked to reconcile their potentially different valuations for the same task between NoCalc and Calc treatments. This exercise aims to reveal which valuations better reflect subjects' welfare-relevant risk preferences. Finally, the subject responds to a few additional questions, including an incentivized choice among four deterministic mirrors.

The complete experimental instructions can be found in Appendix G.

2.3 Implementation Details

The experiment was conducted on Prolific in January 2025. A total of 202 subjects completed the experiment. The experiment was programmed using OTree (Chen, Schonger and Wickens, 2016). Each subject was paid a participation fee of $7 for completing the experiment. With a 20% chance, a subject was also paid the outcome of a randomly chosen task. The median subject spent around 50 minutes on the experiment, and the average total earnings from the experiment were $13.19.

3 Valuations of Lotteries

This section focuses on the valuations of lotteries and mirrors submitted by the subjects. Analysis of the calculations is left to the following sections.

The four panels of Figure 3 show the average absolute valuations for all lotteries and mirrors in both NoCalc and Calc treatments, pooling across all subjects. First, the lottery valuations exhibit substantial unresponsiveness – when the probabilities of lotteries change, the lottery valuations change by a smaller magnitude than the expected values. This pattern of unresponsiveness unifies the classic fourfold pattern documented in Tversky and Kahneman (1992). To see this, note that unresponsiveness usually implies a pull-to-the-center effect in the valuations. When the probability of the non-zero outcome is small, the absolute valuations are greater than the absolute expected values for both gain and loss lotteries, indicating patterns conventionally interpreted as risk-loving preferences for small probability gains, and risk-averse preferences for small probability losses. In contrast, when the probability of the non-zero outcome is large, the relationship between the absolute valuations and the expected values reverses. As a result, the average subject appears to have risk-averse preferences for large probability gains, and risk-loving preferences for large probability losses.

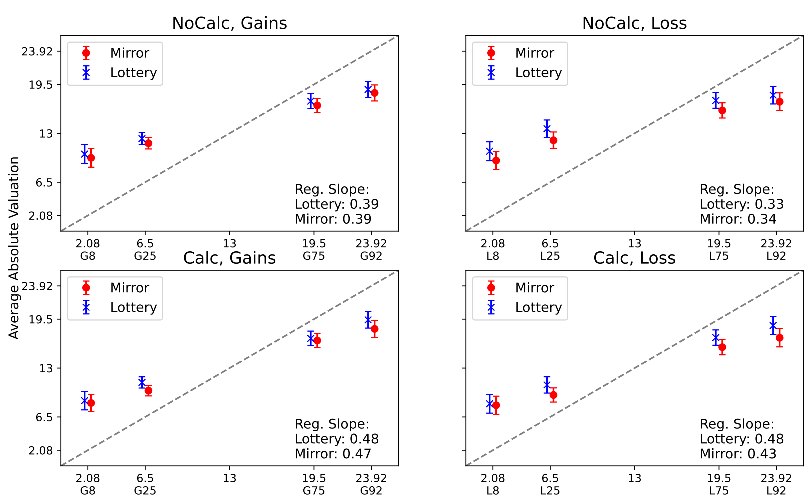


Figure 3: Average valuation of each lottery and their deterministic mirror. 95% confidence intervals are shown with the bars, and the 45-degree line is shown as a dashed line.

Second, replicating Oprea (2024*b*), unresponsiveness also appears in mirror tasks, where an unambiguous correct answer exists that should have entirely eliminated unresponsiveness. Third, the absolute valuations are similar for gain and loss lotteries with the same probability of non-zero outcome, for example G8 and L8.[4] To simplify the analysis, from now on, I treat pairs of gain and loss lotteries with the same probability of non-zero outcome (such as G8 and L8) as the same lottery. That is to say, for a loss lottery, I take absolute values for any characteristics related to it, such as valuation, expected value, and monetary outcomes. In the following text, I will simply use the terminology *valuation* to refer to the absolute valuation, and similarly for other characteristics. Analyzing gain and loss tasks separately does not meaningfully alter any of the results in this paper.

Finally, I examine whether providing access to a calculator affects subjects' lottery and mirror valuations. Figure 3 shows that providing a calculator to subjects does not eliminate the unresponsiveness in their lottery and mirror valuations. However, it is also clear from the slopes shown in the graph that valuations in Calc are somewhat more responsive than those in NoCalc. I will return to the comparison between NoCalc and Calc in Section 5.3 with a more granular analysis focused on subsets of subjects.

4 Descriptions of Calculations

This section first outlines the methodology for analyzing the calculator input data, and then provides descriptive statistics of the data.

4.1 The Data: Calculator Inputs

Since this paper analyzes a novel type of data – the sequence of calculations performed by subjects using the calculator – I first provide a brief overview of the structure of the data.

For each task completed by each subject in the Calc treatment (referred to as a *round*), the computer collects two main pieces of data. First, it collects the subject's *valuation* of this lottery or mirror. Second, it also collects the *calculator input* of the subject. The calculator input is a sequence of numerical expressions typed into the calculator by the subject and evaluated by the calculator. The order of numerical expressions in the calculator input reflects

[4]This pattern repeatedly appears in experiments measuring lottery valuations. See, for example, Tversky and Kahneman (1992, Table 3, page 307) and l'Haridon and Vieider (2019, Figure 2, page 196).

the order by which the subject performs them.

Table 1 visualizes a hypothetical observation of calculator input, where a subject faces the task (G75, Lottery). This calculator input reveals that, when facing the task, the subject first typed $75 \times 26/100 + 25 \times 0/100$ into the calculator and evaluated it, and then did the same for $19.5 - 4.5$. Since the experimental design requires the valuation to appear in the Result column of the last line of the calculator before it is submitted (See descriptions for the Calc treatment in Section 2.2), the result in the last line, 15, is also the valuation.

Line	Numerical Expression	Result
1	$75 \times 26/100 + 25 \times 0/100$	19.5
2	$19.5 - 4.5$	15

Table 1: An example calculator input by a hypothetical subject facing the task (G75, Lottery)

Analyzing the calculator input data involves several important challenges. Most critically, the calculator inputs only provide an *incomplete record* of subjects' decision-making processes when completing valuation tasks. Specifically, the calculator data may fail to capture the complete mapping from task primitives to final valuations for two reasons. First, subjects may rely on process-opaque decision-making that cannot be decomposed into explicit mathematical operations or explicit computational rules. Second, even when the decision-making processes are based on explicit calculation rules, subjects may perform these calculations mentally without entering them into the calculator, making these computational steps invisible in the data. Given these limitations, I do not claim to recover the complete decision-making process for every subject. However, when subjects do perform calculations using the calculator, this data provide valuable insights into the explicit computational approaches they employ when determining their lottery valuations. Moreover, the calculator inputs are high-dimensional objects that require transformation into summary features for quantitative analysis. The features constructed in this process are designed to capture some particular aspects of the calculations, but in the meantime they will inevitably miss some other aspects. To rigorously define the objects and describe the algorithms involved in the analysis, a formal structure of mathematical expressions is needed. An introduction to the formal structure and rigorous descriptions of the algorithms used in this section are provided in Appendix F. In what follows, I rely on examples to illustrate the definitions and algorithms.

4.2 Recovering Symbolic Expressions

While subjects can only calculate numerical expressions in the calculator, their corresponding *symbolic expressions* (in terms of the task primitives) can be recovered to shed light on the calculation strategies by which subjects reach their valuations. Four primitives describe each task: b_1 denotes the number of boxes with non-zero amount of money, m_1 denotes the (absolute) amount in each of these boxes, b_2 denotes the number of boxes with zero amount, and m_2 denotes the amount in each of these boxes. For example, when facing the task (G75, Lottery), where these primitives take values $b_1 = 75$, $b_2 = 25$, $m_1 = 26$, $m_2 = 0$, a subject who calculates 75×26 is actually computing $b_1 \times m_1$. This symbolic expression reveals how subjects utilize task primitives to construct their valuations and facilitates comparisons across tasks with different primitives.

To recover the symbolic expressions, I use a simple match-and-replace algorithm. There are three key features of this algorithm. First, the algorithm processes the calculator input sequentially, from the first to the last line. Second, the algorithm matches the numbers in the numerical expressions against the task primitives. A number that matches a task primitive is replaced with the corresponding symbol. Third, taking into consideration the sequential nature of how subjects perform calculations, starting from the second line, the algorithm also matches the numbers against the set of previous results. A number that matches a previous line result is replaced with the recovered symbolic expression of that line. Any number that does not match any primitive or previous result remains as the same number.

Here, the algorithm is illustrated using the example in Table 1. As the first step, the algorithm matches the numbers in the numerical expression in line 1 against the set of primitives for the task (G75, Lottery): $\{(b_1, 75), (b_2, 25), (m_1, 26), (m_2, 0)\}$.[5] Replacing the matched numbers with their corresponding symbols leads to the symbolic expression of line 1 – $b_1 \times m_1/100 + b_2 \times m_2/100$ – where the number 100 doesn't find a match in the primitives

[5]When implementing the algorithm, the set of primitives that the algorithm matches against is expanded to include common calculation shortcuts that subjects may use. For example, in G75, the primitive set also includes $(b_1 \times m_1/100, 19.5)$ to capture the scenario where a subject calculates the expected value of the lottery in their mind, before using the result of this mental calculation directly in the calculator. For a list of matched shortcuts, see Appendix F.

and is left intact. Next, the algorithm moves to line 2. The number 19.5 does not match any task primitive, but it matches the result of line 1. Thus, the algorithm replaces the number 19.5 with the recovered line 1 symbolic expression ($b_1 \times m_1/100 + b_2 \times m_2/100$). In this way, the algorithm recovers the line 2 symbolic expression – $b_1 \times m_1/100 + b_2 \times m_2/100 - 4.5$. The results are shown in Table 2.

Line	Numerical Expression	Result	Symbolic Expression
1	$75 \times 26/100 + 25 \times 0/100$	19.5	$b_1 \times m_1/100 + b_2 \times m_2/100$
2	$19.5 - 4.5$	15	$b_1 \times m_1/100 + b_2 \times m_2/100 - 4.5$

Table 2: Symbolic expressions of the example in Table 1, where a hypothetical subject faces the task (G75, Lottery).

The incomplete record limitation of the calculator input data can be illustrated by comparing the two examples Table 2 and Table 3. In line 1, the subject in Table 3 performs the same expected value calculation as Table 2. However, suppose that after computing the expected value, instead of explicitly subtracting of 4.5 from the expected value as in Table 2, the subject performs this fairly simple calculation in their head, and types 15 directly into the second line and submits this as their valuation. Now, in Table 3, the number 15 in line 2 cannot be matched to any task primitives or previous results. As a result, the symbolic expression is simply the constant function 15, and the connection between the number 15 and the task primitives cannot be seen from this recovered symbolic expression.

Line	Numerical Expression	Result	Symbolic Expression
1	$75 \times 26/100 + 25 \times 0/100$	19.5	$b_1 \times m_1/100 + b_2 \times m_2/100$
2	15	15	15

Table 3: An example calculator input by a hypothetical subject facing the task ($G75$, *Lottery*).

4.3 Procedures and Base Terms

To address the high-dimensionality of the calculator input data, I develop two complementary features of calculator inputs to facilitate the analysis. Both features rely on the recovered symbolic expressions, but they differ in terms of their focus. First, I construct the *procedure* to capture how the subject *maps task primitives to their final valuation.* Second, I construct the set of *base terms* to summarize *all* functional forms of task primitives a subject calculates in a round.

Procedures For a round, its procedure is defined as the recovered symbolic expression of the last line of the calculator input.[6] Since the experimental design requires the valuation to appear in the Result column of the last line of the calculator before it is submitted, the procedure is the function mapping task primitives to the final valuation. For example, in the calculator input shown in Table 2, its procedure is $b_1 \times m_1/100 + b_2 \times m_2/100 - 4.5$.

Ideally, the definition of procedures attempts to capture the function that the subject uses to map the primitives of the task to their valuations. However, the example illustrated in Table 3 has shown a potential pitfall of this attempt. In Table 3, the procedure itself, being the constant function 15, is silent on the fact that the subject also calculates the expected value of the lottery, and may have used the calculated expected value in a way that is not captured by the calculator input data to construct their valuation. To address this problem, I introduce base terms to complement procedures in the analysis of calculator inputs.

Base Terms and Base Term Sets I decompose a calculator input into its base terms: All terms that appear in the symbolic expressions of the calculator input, with their numerical factors dropped. The process of identifying base terms involves three steps:

1. First, I break down each symbolic expression in the calculator input into terms.[7] For example, in the symbolic expression $b_1 \times m_1/100 + b_2 \times m_2/100$, there are two terms: $b_1 \times m_1/100$ and $b_2 \times m_2/100$.

2. Next, I generate the base term of each term by dropping all its numerical factors. For example, starting from the term $b_1 \times m_1/100$, dropping the numerical factor 1/100 from it generates its base term $b_1 \times m_1$. If a term contains only numerical factors and no symbolic factors (Examples: (i) 4; (ii) 4×2), its base term is defined as C.

[6] More precisely, the procedure is the *function of task primitives* represented by the recovered symbolic expression of the last line of the calculator input. This lengthy definition emphasizes the fact that a procedure is a function, not an expression. In other words, two expressions with different syntaxes but representing the same function, for example, $2 \times b_1 \times m_1/200$ and $(b_1 \times m_1)/100$, should be viewed as the same procedure.

[7] The concept of terms, and by extension base terms, suffers from an indeterminacy problem with syntactically different but mathematically equivalent expressions. For example, the mathematically equivalent expressions $b_1 \times m_1/100 + b_2 \times m_2/100$ and $(b_1 \times m_1 + b_2 \times m_2)/100$ lead to different base terms. To solve this problem, I first expand all the products in all expressions by applying the distributive law of multiplication ($a \times (b + c) = a \times b + a \times c$), wherever applicable. This way, I transform the original symbolic expression into its distributed form expression. The base term set of a calculator input is defined as the collection of all base terms that appear in any of its distributed form expressions. Using distributed form expressions solves the aforementioned indeterminacy problem and generates the same set of base terms for $b_1 \times m_1/100 + b_2 \times m_2/100$ and $(b_1 \times m_1 + b_2 \times m_2)/100$.

3. Finally, I define the *base term set* of a calculator input as the collection of all base terms that appear in any of its symbolic expressions.

This definition is illustrated again using the example in Table 1. In the symbolic expression of line 1 ($b_1 \times m_1/100 + b_2 \times m_2/100$), two base terms appear: $b_1 \times m_1$ and $b_2 \times m_2$, whereas in the symbolic expression of line 2 ($b_1 \times m_1/100 + b_2 \times m_2/100 - 4.5$), a total of three base terms appear: C in addition to $b_1 \times m_1$ and $b_2 \times m_2$. Therefore, the base term set of this example calculator input is $\{b_1 \times m_1, b_2 \times m_2, C\}$.

The base terms provide a summary of the functional forms of *all calculations*, while the procedures capture the exact functional form of the *final valuation*. These two features both reduce the dimension of the calculator input, and in the meantime complement each other. First, procedures complement base terms by showing how the base terms are combined to form the valuations. Second, since the procedures lose any information over the calculations that cannot be directly connected to the valuation, base terms complement procedures by preserving the information in these calculations. For instance, though the examples in Table 2 and Table 3 generate different procedures, they generate the same base term set $\{b_1 \times m_1, b_2 \times m_2, C\}$, which emphasizes the similarity of these two calculator inputs.

Procedure Groups and Base Term Groups For parsimony and interpretability, I categorize base terms into five groups, listed in Table 4. Beyond their intuitive appeal, this categorization is also supported by fitting an unsupervised machine learning classification model using the calculator inputs.[8] Parallel to the base term groups, I also classify all procedures into five groups.

4.4 Descriptions of Calculator Inputs

Now, I provide descriptions of the calculator inputs appearing in the experiment. First, in 75% of all rounds, some explicit calculations are performed, while in the remaining 25% of all rounds, no explicit calculation is performed at all, and the subjects simply type a number

[8] I use Latent Dirichlet Allocation (LDA, Blei, Ng and Jordan, 2003) to find latent *topics* from calculator inputs, and group base terms by the topic that they are strongly associated with. The five groups of base terms listed in the main text are each associated with an individual topic. This exercise draws an analogy between calculator inputs and text documents in natural language. LDA is a popular unsupervised technique in natural language processing to find latent semantic topics from text documents. Details of this unsupervised machine learning approach can be found in Appendix B.

Group	Base Terms	Procedures
Expected value	$b_1 \times m_1, b_2 \times m_2$	$b_1 \times m_1/100$ or $(b_1 \times m_1 + b_2 \times m_2)/100$
Number	C	Constant functions (e.g., 5)
Linear box	b_1, b_2	$\theta_0 + \theta_1 \times b_1 + \theta_2 \times b_2$, where $\theta_0, \theta_1, \theta_2$ are constants
Linear money	m_1, m_2	$\gamma_0 + \gamma_1 \times m_1 + \gamma_2 \times m_2$, where $\gamma_0, \gamma_1, \gamma_2$ are constants
Non-linear	Anything else	Anything else

Table 4: Classification of base terms and procedures into five groups

into the calculator and submit this number.

The next question I aim to answer is: What calculations do subjects perform when they value lotteries and mirrors? Particularly, besides the calculations of the expected value, does there exist any other functional form of task primitives that is repeatedly calculated by different subjects? To answer this question, in Panel A of Table 5, I list all non-number procedures that appear in more than 0.5% of all rounds, in addition to their shares in the two task types separately. Except for the expected value procedures and a few linear money procedures, other non-number procedures appear in only a tiny share of rounds.

Panel A: Procedures			Panel B: Procedure Groups		
	lottery	mirror		lottery	mirror
$b_1 \times m_1/100$	29.0%	32.9%	expected value	34.7%	37.0%
m_1	11.3%	6.6%	number	31.9%	34.8%
$b_1 \times m_1/100 + b_2 \times m_2/100$	5.7%	4.1%	linear money	20.6%	14.8%
$m_1/2$	0.8%	2.8%	nonlinear	9.0%	9.4%
m_2	2.1%	0.9%	linear box	3.7%	4.1%
$b_2 \times m_1/100$	1.5%	1.0%			
$m_1/4$	1.0%	0.4%			
$b_1 \times m_1/200$	0.4%	1.0%			
$m_1 + m_2$	0.8%	0.6%			
$3 \times b_1/10$	0.7%	0.6%			
$100/b_1$	0.6%	0.7%			

Table 5: Frequencies of procedures and procedure groups. Panel A: Shares of all non-number procedures that appear in more than 0.5% of all rounds (lottery and mirror). Panel B: Shares of all five procedure groups (see the text for the definition).

Looking at the base terms paints the same picture. Across the 3636 Calc rounds, only 44 distinct base terms appear, and only 8 of these appear in more than 1% of rounds (listed in Panel A of Table 6). These most frequently used base terms often have interpretable functional forms: the components of expected value calculations ($b_1 \times m_1$ and $b_2 \times m_2$) and the linear terms m_1, m_2, b_1, b_2. The number term C, which represents unmatched number terms in the calculations, appears in 38.0% (36.9%) of lottery (mirror) rounds.[9] All unlisted base terms together appear in only 5.9% (6.7%) of lottery (mirror) rounds. In other words, even when given the flexibility to perform any calculation, subjects overwhelmingly restrict themselves to expected value or linear functions of task primitives, while other functional forms are only used sporadically. In addition, in 85.0% of rounds, subjects employ base terms from only a single group, which indicates that subjects typically do not combine multiple base term groups to construct more complex valuation rules. These results rule out the possibility that a non-negligible fraction of subjects develops highly complicated non-EV valuation rules that are explicitly implementable in the calculator.

Panel A: Base Terms			Panel B: Base Term Groups		
	lottery	mirror		lottery	mirror
$b_1 \times m_1$	40.6%	43.7%	expected value	40.7%	43.7%
C	38.0%	36.9%	number	38.0%	36.9%
m_1	20.1%	18.3%	linear money	23.4%	20.5%
$b_2 \times m_2$	7.0%	6.4%	linear box	7.8%	7.3%
b_1	6.5%	5.8%	nonlinear	7.1%	7.0%
m_2	4.9%	3.6%			
b_2	4.1%	3.4%			
$b_2 \times m_1$	1.9%	1.4%			
All others	5.8%	6.4%			

Table 6: Fractions of rounds where each base term (group) appears. Panel A: Fractions of all base terms that appear in more than 1% of all rounds (lottery and mirror). Panel B: The fractions of all five base term groups (see the text for the definition).

Result 1. *Subjects predominantly employ one of three main groups of calculations when valuing lotteries: (1) expected value; (2) number; or (3) linear money. These three groups account for the vast majority of calculations, other calculations are rare, and subjects typically use a single group of calculations per round.*

Moreover, across lottery and mirror tasks, the shares of procedures and base terms are similar. The finding micro-founds the similar lottery and mirror valuations documented in

[9] Most of these number terms consist of only one number factor, such as 4, as opposed to a few number factors multiplied together, such as 4×2. More specifically, if I only look at the frequency of terms with only one number factor, they appear in 28.6% of lottery rounds and 27.9% of mirror rounds.

Oprea (2024*b*) and in Section 3 of this paper. Examples of calculator inputs appearing in the data can be found in Appendix C.

4.5 Within-Subject Stability of Calculator Inputs

Next, I ask the following question: Does the same subject use stable calculations across tasks? There are two facets of this stability. First, I examine *within-task-type* stability of procedures: Does the same subject use the same procedure across tasks within the same task type? For each combination of subject and task type, I identify the *modal procedure*: The most frequently occurring procedure within that subject's calculator inputs for all rounds of that task type. I then compute what fraction of that subject's rounds use their modal procedure. Taking the median of these fractions across subjects shows that the median subject uses their modal procedure in 5 out of 9 rounds for both lottery and mirror tasks.

Using exactly the same procedure for multiple tasks requires that every step in the valuation process is explicitly performed in the calculator. If a subject performs part of their valuation process implicitly in their mind, their procedures as defined here will differ across rounds, but the actual valuation processes may still be similar. This argument demonstrates the value of conducting an additional analysis using the coarser notion of procedure groups, as opposed to raw procedures. Using the same procedure group in two rounds indicates a generally similar approach to constructing the valuations.[10] An analogous analysis as what has been done above for procedures reveals that the median subject uses their modal procedure group in 8 out of 9 rounds for both lottery and mirror tasks. This analysis strongly suggests that the majority of subjects maintain a fairly stable procedure group for a given task type.

Second, I examine *between-task-type* stability: Does the same subject use the same procedure when valuing *a lottery and its corresponding mirror*? I refer to all lottery-mirror round pairs where the same subject faces a lottery and its matched mirror as *within-subject pairs*. I find that 44.9% of within-subject pairs have identical procedures, while as a benchmark,

[10] For example, in the extreme case where a subject's valuation process is fully process-opaque and indescribable in the calculator, the subject will simply submit a number as their valuation without performing any calculation in the calculator. In this case, all the subject's procedures will be in the number group, indicating that the subject uses a similar (process-opaque) way to approach different tasks, despite the fact that the procedures are different.

the analogous figure for all lottery-mirror round pairs (including pairs across different subjects) is only 11.2%. Looking at the coarser notion of procedure groups instead of the procedures, among within-subject pairs, 68.9% have identical procedure groups (benchmark using all lottery-mirror pairs: 28.0%). The result strongly suggests that subjects typically use similar approaches to valuing a lottery and its corresponding mirror.[11]

Although I have focused on the stability of procedures in this section, the analysis using base terms leads to identical results.

Result 2. *The procedures and base term sets are generally stable within-subject either within lottery tasks, within mirror tasks, or between lottery and mirror tasks.*

Moreover, among the within-subject pairs where lottery and mirror procedures do differ, there is only a weak tendency for subjects to switch from non-EV procedures in lotteries to EV procedures in mirrors. Appendix Table A.1 shows the joint distribution of lottery procedure group and mirror procedure group among all within-subject pairs. In 8.0% of all within-subject pairs, the subjects switch from an EV-group procedure in the mirror task to a non-EV-group procedure in the lottery task, while in 5.8% of all these pairs, the subjects switch in the opposite direction. The difference between the frequencies of these two opposite switch directions is economically small.

5 Linking Calculator Inputs to Valuations

The next question I study is the connection between the calculator inputs and valuations. Section 5.1 looks at the round-level predictive power of calculator inputs for valuations. I categorize rounds by the base terms in their calculator inputs, and then examine the distributions of valuations conditional on employing different base terms. Next, in Section 5.2, I turn to the subject level and aggregate across rounds. I assign types to subjects based on their calculator inputs in lottery tasks, and analyze the link between the types of subjects and their responsiveness in the Calc treatment. Section 5.3 examines the link between the types of subjects and their responsiveness in the NoCalc treatment.

[11]The between-task-type stability is not entirely driven by the prevalence of the expected value and the number group procedures. Appendix Table A.1 shows the fractions of within-subject pairs that have identical procedure groups, conditional on the procedure group in the lottery task. All these conditional fractions are substantially higher than the benchmark of 28.0%.

5.1 Round-Level Calculator Inputs and Valuations

Does the calculator input in a round predict the valuation in the same round? Figure 4 shows the cumulative distribution functions (CDF) of the valuations in lottery tasks in the Calc treatment, where each panel plots the CDF conditional on employing a group of base terms. When multiple groups of base terms are used within a single round, the valuation from that round appears in the CDF for each applicable group.

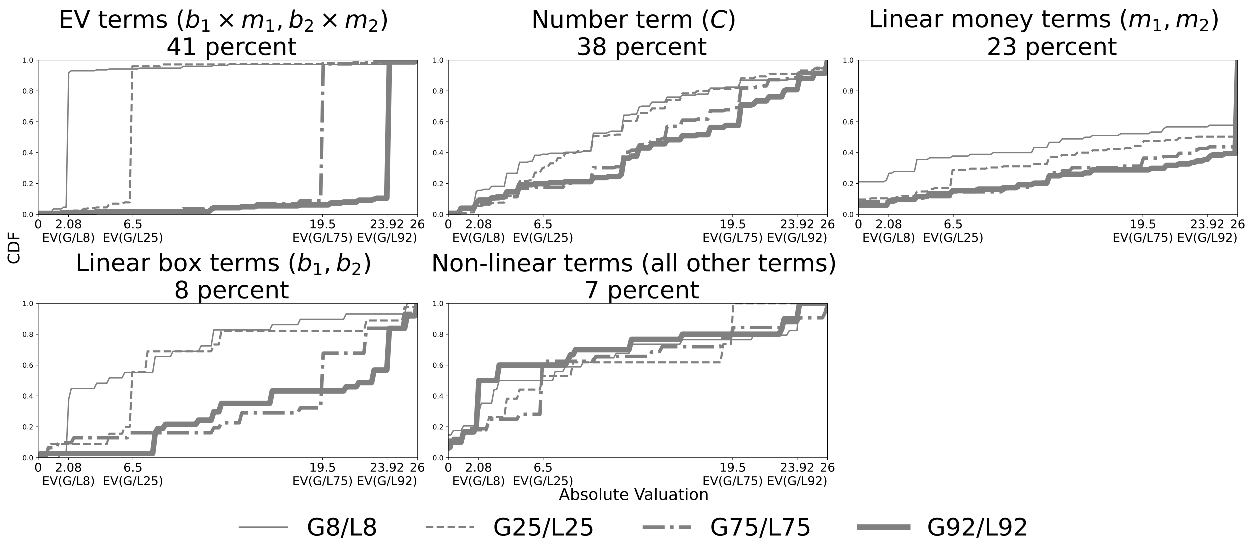


Figure 4: Distributions of lottery valuations in the Calc treatment, conditional on employing each group of base terms.

The data reveal that when subjects employ EV terms in a round, their resulting valuations tend to be risk neutral. In principle, even when EV terms are employed, valuations could still deviate from expected values through two mechanisms. First, since numerical factors are dropped when constructing base terms, a subject who calculates, for example, $0.9 \times b_1 \times m_1/100$ would produce a valuation that differs from the expected value. Second, since Figure 4 shows data conditional only on the presence of EV terms, subjects may simultaneously employ additional non-EV terms in their calculations. For instance, a subject might calculate $b_1 \times m_1/100 - 1$, and this round would still be included in the EV terms panel of Figure 4 despite the subtraction of a constant. Nevertheless, the data show that when EV terms are employed, valuations remain predominantly clustered around the expected value.[12]

[12]Appendix Figure A.2 displays the CDF of lottery valuations separated into three disjoint groups of rounds: 1) rounds employing only EV terms; 2) rounds employing only non-EV terms; and 3) rounds employing

Result 3. *At the round-level, EV terms are predictive of lottery valuations that are responsive and are tightly concentrated around the expected value. In contrast, non-EV terms are predictive of valuations that are unresponsive and highly dispersed.*

Although the previous analysis has focused on lotteries, quantitatively similar results also appear for mirrors: EV terms in a mirror round predict responsive and concentrated mirror valuations, while non-EV terms predict the opposite. Appendix Figure A.3 replicates Figure 4 using the mirror tasks.

5.2 Subject Types and Their Responsiveness

Do the calculator inputs of a subject predict their responsiveness? I categorize subjects based on their *modal base term* in lottery tasks in the Calc treatment. For a subject, their modal base term is the base term that is used in the highest number of rounds, among all lottery rounds of this subject. Their *type* is the base term group to which their modal base term belongs.[13] The types are a simple but powerful summary of the calculator inputs at the subject-level, due to the within-subject stability of calculations documented in Result 2. The five base term groups correspond to five subject types: 1) Expected value; 2) Number; 3) Linear box; 4) Linear money; 5) Non-linear. To validate that the subject within a type indeed primarily used the base term group associated with that type, Table 7 shows the fractions of rounds where each base term group appears (similar to Panel B of Table 6), conditional on each subject type. All types use their associated base term group in more than 80% of rounds, and non-associated base term groups are only used sporadically.

I construct a measure of an individual subject's responsiveness using the regression slope of valuations on expected values. Specifically, I run the regression

$$|\text{valuation}| = \alpha + r\ |\text{expected value}| + \epsilon \tag{1}$$

using all lottery tasks completed by the subject in the Calc treatment. The subject's

both EV and non-EV terms. The figure demonstrates that when valuations do deviate from expected values despite the presence of EV terms, these deviations are primarily attributable to the concurrent use of non-EV terms rather than to the dropped numerical factors.

[13]The categorization is robust to a machine learning-based categorization based on the estimated unsupervised topic model, mentioned in Footnote 8. See Appendix B for more details.

	Subject Type				
Base Term Group	expected value	number	linear money	linear box	nonlinear
expected value	**92.2%**	5.7%	6.2%	40.0%	12.7%
number	5.5%	**89.0%**	14.4%	15.6%	3.2%
linear money	4.6%	15.8%	**86.3%**	11.1%	22.2%
linear box	2.5%	4.8%	3.9%	**88.9%**	1.6%
nonlinear	8.1%	1.8%	1.3%	6.7%	**81.0%**

Table 7: Fractions of lottery rounds where each base term group (the row) appears, conditional on subject type (the column)

responsiveness in lottery tasks is the estimated coefficient r. A responsiveness of 0 indicates complete unresponsiveness – on average, the valuations do not change with the expected values. On the other hand, a responsiveness of 1 indicates complete responsiveness – the valuations change by the same magnitude as the expected values. This measure of an individual subject's responsiveness can be expanded to the mirror tasks and the NoCalc treatment by the slopes of analogous regressions.

Figure 5 shows the histograms of individual responsiveness in lottery tasks in the Calc treatment, first for all subjects and then separately for each subject type. The responsiveness is censored at a lower bound of -0.2 in the graphs for improved visibility. From the upper-left panel encompassing all subjects, it is immediate to see that responsiveness has a bimodal distribution – most subjects concentrate around complete responsiveness ($r = 1$) and complete unresponsiveness ($r = 0$), and only a small fraction of subjects are in the middle range. The average responsiveness in the subject population is 0.48.

The distributions of responsiveness for different subject types reveal substantial heterogeneity beneath the bimodal aggregate distribution. These two distinct modes correspond to two broad groups of subjects. The EV-type subjects (38.1% of the population) tend to exhibit responsiveness of nearly one (average = 0.85), which implies their lottery valuations closely track expected values. For example, these subjects value lotteries G8/L8 (with $|EV| = 2.08$) at an average of \$3.36, while valuing lotteries G92/L92 (with $|EV| = 23.92$) at an average of \$21.48. In contrast, non-EV-type subjects (61.9% of the population) exhibit much lower responsiveness (average = 0.25). Their valuations increase only minimally as probability increases: their average valuation of lotteries G8/L8 is \$11.40, while their average

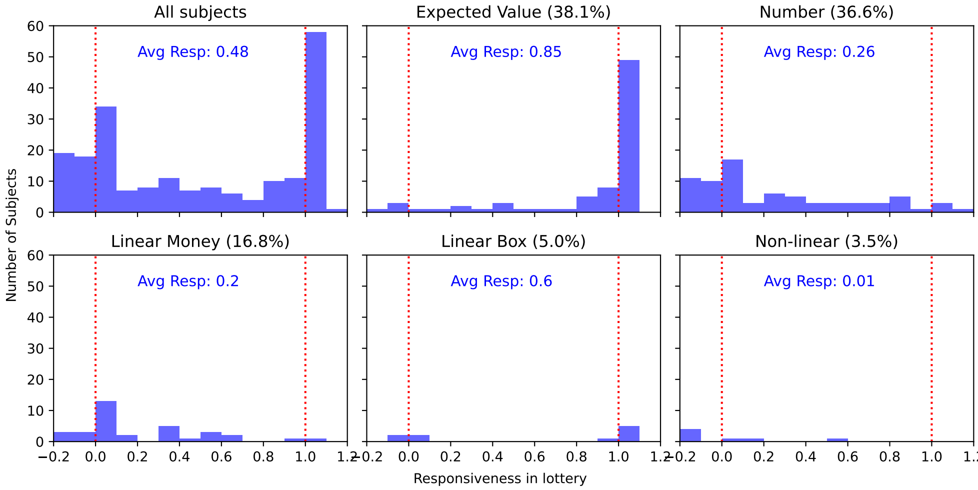


Figure 5: Histograms of individual responsiveness in lottery tasks in the Calc treatment

valuation of lotteries G92/L92 is only \$16.38, despite the large difference in expected values. In aggregate, this stark difference between EV-type and non-EV-type subjects produces the bimodal distribution of responsiveness observed in the overall sample.

Most strikingly, a substantial proportion of non-EV-type subjects exhibit responsiveness close to or even below zero. Among all non-EV-type subjects, 26.4% exhibit responsiveness between 0 and 0.1, and another 26.4% exhibit responsiveness below 0. For the non-EV-type subjects, their extreme unresponsiveness and prevalent violations of monotonicity seriously challenge the risk preferences interpretation of unresponsiveness.[14]

Moreover, although subject types are constructed only using calculator inputs in lottery tasks, they can also predict the responsiveness in mirror tasks. Appendix Figure A.1 shows the histograms of individual responsiveness in mirror tasks in the Calc treatment by subject type. For all types, the responsiveness in mirror tasks is quantitatively similar to that in lottery tasks. The EV-type subjects submit responsive mirror valuations (average = 0.82),

[14] It is worth pointing out that the fraction of subjects who are extremely unresponsive in this study is at the higher end of the literature, but not unprecedented. In this study, 37.1% of all subjects exhibit lottery responsiveness below 0.1 in the NoCalc treatment, and 35.1% in the Calc treatment. To facilitate comparison with other studies, it is important to note that many experiments ask subjects to value binary lotteries that vary in both probabilities and monetary outcomes, whereas the responsiveness measure in this paper primarily captures sensitivity to probability changes. Accordingly, I define responsiveness in other studies at the subject-outcome pair level, where a given subject values lotteries with fixed monetary outcomes but varying probabilities. Using this definition, the two experiments in Enke and Graeber (2023) show that 18.6% and 32.7% of subject-outcome pairs, respectively, exhibit responsiveness below 0.1. l'Haridon and Vieider (2019) report that 10.2% of subject-outcome pairs have responsiveness below 0.1, McGranaghan et al. (2024) report 10.1%, and across three experiments in Bruhin, Fehr-Duda and Epper (2010), this fraction ranges from 9.6% to 13.7%.

but interestingly, their mirror valuations are on average slightly *less* responsive than their lottery valuations. The non-EV-type subjects submit highly unresponsive mirror valuations, and their mirror responsiveness is similar to that in lottery tasks.

Result 4. *In both lottery and mirror tasks, EV-type subjects tend to be highly responsive, while non-EV-type subjects tend to be highly unresponsive. In aggregate, the distribution of unresponsiveness is bimodal.*

The low responsiveness of the non-EV-types is not entirely driven by subjects submitting identical valuations across many tasks. Appendix Figure A.4 reproduces Figure 5 excluding those subjects who submit the exact same valuation to no fewer than 7 out of a total of 9 lottery rounds. The apparent bimodal distribution of responsiveness is robust to excluding these subjects, and the remaining non-EV-types still exhibit very low responsiveness.

5.3 Types and Responsiveness in the NoCalc Treatment

After documenting the responsiveness of each subject type in the Calc treatment, I now turn to their responsiveness in the NoCalc treatment. I study this with two purposes. First, I examine to what extent the calculator design is externally valid – capturing the approaches subjects would naturally take facing decisions under risk in the absence of the calculator. Second, if for some types of subjects the valuations do differ between treatments, I study how the valuations change between treatments, and which set of valuations better reflect their risk preferences.

Figure 6 shows the distributions of individual responsiveness in lottery tasks in the NoCalc treatment by subject type. First, even though types are constructed only using calculator input data from the Calc treatment, they are still predictive of the responsiveness in the NoCalc treatment – the EV-types exhibit higher responsiveness than the non-EV-types in the NoCalc treatment. Second, the aggregate distribution of responsiveness in the subject population differs between treatments – there are relatively fewer subjects with high responsiveness (near 1) and relatively more subjects with intermediate responsiveness in NoCalc. Third, this difference is attributable solely to the EV-types. The EV-types exhibit lower responsiveness in the NoCalc treatment when compared with the Calc treatment, while the other types

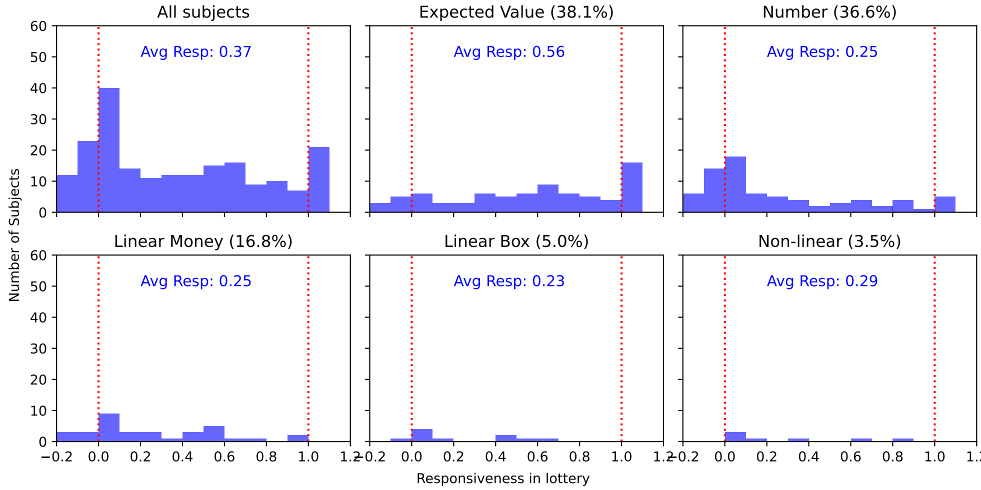


Figure 6: Histograms of individual responsiveness in lottery tasks in the NoCalc treatment

show similar responsiveness across treatments.[15] Using the Kolmogorov-Smirnov (K-S) test to compare the distributions of valuations in the two treatments separately for each subject type and task confirms this result. After applying the Holm-Bonferroni method to control the familywise error rate, the K-S tests reject the null hypotheses of equal distributions for EV-types in 7 of the 8 lottery tasks, while the K-S tests fail to reject the null hypotheses for 31 out of the 32 (8 tasks $\times$ 4 non-EV-types) tests involving non-EV-types.

Result 5. *Comparing across treatments reveals an overall shift toward higher responsiveness in the Calc treatment. This shift is entirely driven by EV-type subjects, while non-EV-types submit statistically indistinguishable valuations across treatments.*

Overall, these observations indicate that, for non-EV-types, the calculator design seems to well capture the natural approaches subjects would take facing decisions under risk, while the EV-types take a different approach in the presence of a calculator. For the EV-types, a

[15]Comparing the Calc treatment with the NoCalc treatment, the two patterns of (1) a slight increase of aggregate responsiveness and (2) a sharp increase of subject masses around complete responsiveness are broadly consistent with previous studies. Beauchamp et al. (2020) conduct lottery valuation experiments with a treatment condition where they explicitly provide the expected value on subjects' screens when they value the lotteries. I conduct a re-analysis of Beauchamp et al.'s (2020) data by constructing responsiveness at the subject-outcome pair level (See Footnote 14 for the definition of subject-outcome pairs). Providing subjects explicitly with the expected values of the lotteries increases the fraction of subject-outcome pairs with $r > 0.9$ from 24.4% to 33.7%, and the average responsiveness from 0.57 to 0.65. Moreover, Gao and Garagnani (2025) provide subjects with a calculator when they value lotteries, without tracking the calculations. They show the calculator leads to an increase of risk-neutral choices, but no decrease of choices violating first order stochastic dominance. This is consistent with my data, since the calculator only changes the valuations of the EV-types, who become almost risk neutral with the calculator, and tend not to make dominated choices in the NoCalc treatment.

natural follow-up question is: Which valuations better represent their welfare-relevant risk preferences? The reconciliation stage speaks to this question, where the subjects are given a chance to reconcile their inconsistent valuations in the NoCalc and Calc stages for the same task. In the reconciliation stage, the EV-type subjects revise their valuation in the NoCalc treatment to the one in the Calc treatment more than three times as frequently as they do the opposite. In other words, their almost risk-neutral valuations in Calc better represent their risk preferences than their moderately unresponsive valuations in NoCalc. See Appendix E for more details from the analysis of the reconciliation stage. For the EV-types, the between-treatment change of valuations and the reconciliation stage results have important theoretical implications, which I will return to in Section 6.

6 What Drives Unresponsiveness?

What drives the unresponsiveness observed in lottery valuations? The key advantage of my experimental design is that the calculator input data can help distinguish between theoretical mechanisms that are indistinguishable using only valuations. This section uses the joint patterns of the calculator inputs and valuations to examine the mechanisms behind unresponsive lottery valuations through three perspectives. First, I briefly revisit the debate of whether the unresponsive lottery valuations mainly reflect risk preferences or cognitive complexity. Second, to the extent that complexity can explain unresponsiveness, the exact nature of this complexity remains understudied. I study the nature of this complexity by focusing on two broad camps of complexity, implementation costs and incomplete understanding. Third, I examine whether two specific theories – probability weighting and attribute substitution – can explain the linear money calculations.

6.1 Risk Preferences or Complexity-Driven Mistakes? A Revisit

A central question in decision-making under risk concerns whether observed unresponsive lottery valuations primarily reflect underlying risk preferences (economic agents' welfare-relevant ordering) or complexity-driven mistakes (costs or difficulty to make decisions according to the risk preferences), which is a distinction with profound implications and that has sparked a recent debate (Oprea, 2024*b*, Banki et al., 2025, **?**, Li et al., 2025, Wakker, 2025).

If unresponsiveness stems mainly from risk preferences, it should guide both economic welfare analysis and policy design. Conversely, if it reflects primarily complexity-driven errors, this calls for careful treatment of valuation data in welfare analysis and suggests policy interventions to improve decision quality.

Using only the lottery valuation data – without relying on information from the calculations or mirror tasks – I document a distinctive empirical pattern that informs this question: the bimodal distribution of responsiveness when subjects are provided with calculators. As becomes evident from comparing Figure 5 with Figure 6, the probability mass in the middle range of responsiveness diminishes substantially after providing subjects with calculators, creating two distinct modes. This bimodal pattern offers important insights into the interpretation of measured responsiveness.

For both modes, the evidence suggests that the unresponsiveness these subjects exhibit *in the NoCalc treatment* (which resembles the experiments measuring risk attitudes in the literature) is unlikely to be solely driven by risk preferences. On the one hand, for the highly responsive mode (largely overlapping with the EV-types), the reconciliation stage suggests that these subjects' preferences are approximately risk-neutral. Therefore, their unresponsiveness in the NoCalc treatment largely stems from non-risk preference factors. On the other hand, for the subjects in the extremely unresponsive mode (defined as $r < 0.1$, mostly consisting of the non-EV-types), while risk preferences may have potentially contributed to their unresponsiveness, the magnitude of unresponsiveness observed exceeds what can plausibly be attributed to risk preferences alone, though the relative importance of complexity and risk preferences remains an active area of investigation. Therefore, the only subjects whose unresponsive NoCalc valuations could be plausibly attributed solely to risk preferences are those moderately responsive in the Calc treatment, lying between the modes.

To quantify how much of the aggregate unresponsiveness is due to the two modes and how much is due to the moderately responsive, I decompose the aggregate unresponsiveness into the contribution of each subset of subjects. The contribution of any subset G is defined as $C(G) := \frac{\sum_{i \in G}(1-r_i)}{\sum_{j \in S}(1-r_j)}$,[16] where S is the set of all subjects, and r_i is the responsiveness of

[16]In my data where all subjects perform the same set of tasks, it can be shown that $\frac{\sum_{i \in G} r_i}{|G|}$ (i.e., the *average* individual responsiveness in subset G) is the regression slope of Equation (1) using the data from all

subject i as defined in Equation (1). This decomposition exercise reveals that the mode exhibiting high responsiveness in the Calc treatment (defined as $r > 0.9$, 34.7% of subjects) contributes 21.1% to the aggregate unresponsiveness in the NoCalc treatment, the mode exhibiting extreme unresponsiveness (defined as $r < 0.1$, 35.1% of all subjects) contributes 49.8%, and the rest only contribute 29.0%.

In summary, most of the aggregate unresponsiveness is attributable to either of the modes, and only a minority can be attributed to the moderately responsive subjects. In other words, most of the aggregate unresponsiveness cannot be solely attributed to risk preferences, and complexity plays a significant role in the elicited lottery valuations.

6.2 Implementation Costs Or Incomplete Understanding?

To the extent that complexity can explain unresponsiveness, the exact nature of this complexity remains understudied. I classify complexity into two broad camps, and seek evidence in the data that supports or contradicts each camp. The camp of *implementation costs* attributes unresponsiveness to costs of implementing optimal procedures despite awareness of them, while the camp of *incomplete understanding* attributes it to a lack of complete conceptual understanding of the lotteries. The two camps are not strictly mutually exclusive, but are useful as a broad classification of the mechanisms behind unresponsiveness (Handel and Schwartzstein, 2018).

The conceptual distinctions between implementation costs and incomplete understanding are of fundamental interest – if the implementation costs prevail, unresponsive lottery valuations documented in the lab experiments will have limited predictive power for high-stakes real-life decisions under risk, since the implementation costs can be overcome with the high-stakes, while incomplete understanding indicates the opposite, and calls for policy solutions to help people make decisions under uncertainty. See more discussions on this distinction in Handel and Schwartzstein (2018) and Enke et al. (2023).

Implementation Costs The theory of implementation costs explains unresponsive lottery valuations by the costs of implementing valuation procedures that represent their risk

subjects in any $G \subseteq S$ (i.e., the *aggregate* responsiveness in subset G). This gives $C(G)$ its interpretation as the contribution of subset G to the aggregate unresponsiveness. In other words, $C(G)$ measures how much the aggregate unresponsiveness would decrease, if all subjects in G exhibited complete responsiveness.

preferences (see, e.g., Payne, Bettman and Johnson, 1988, Oprea, 2024*a*). For example, it might be costly for economic agents to calculate the expected values of lotteries. As a result, even if some agents are risk neutral and understand that calculating the expected values leads to their preferred lottery valuations, these subjects may resort to less costly calculations and, in turn, submit lottery valuations that are unresponsive. When valuing mirrors, these agents face analogous implementation costs in calculating the expected values, and may similarly choose simpler calculations that generate valuations similar to those of lotteries.

The theory of implementation costs offers two predictions testable in the current study. First, decreasing the implementation costs should lead to more responsive valuations. This prediction is verified in Result 5, where I document that the EV-type subjects submit much more responsive valuations in the Calc treatment, where implementation costs for explicit computational procedures are arguably smaller, than in the NoCalc treatment.

Subject Type	Lottery Avg Length	Mirror Avg Length
Expected value	11.3	11.5
Number	4.4	5.4
Linear money	4.5	5.1
Linear box	11.0	9.9
Nonlinear	13.0	12.7

Table 8: Length of Calculator Inputs (in Characters) by Subject Type and Task Type

The second prediction of implementation costs is that the calculations generating unresponsive valuations should involve smaller implementation costs than those generating responsive valuations. I test this prediction by constructing a proxy of the implementation costs: the *length* of calculator inputs, which is defined as the total number of characters (digits, operation signs, and decimal points) in all numerical expressions in the calculator input. This metric directly captures implementation costs in our experimental environment since, by design of the calculator, each character requires a separate operation – either a button click or a keystroke – to type. For example, calculating $75 \times 26/100$ requires nine operations, while entering 15 requires only two.

Table 8 shows the average length of calculator inputs by subject type and task type. The data reveal a clear pattern: number-type and linear money-type subjects consistently perform less costly calculations than the EV types, for both lottery and mirror tasks. This aligns with the predictions of implementation costs. However, the current experimental design cannot establish a *causal* link between implementation costs and the choice of shorter calculator inputs, and in turn, the unresponsive valuations they generate.

Result 6. *There is strong evidence that implementation costs play a significant role in the unresponsiveness exhibited by EV-type subjects in the NoCalc treatment. For number and linear money types, the evidence supporting the roles of implementation costs is suggestive and less definitive.*

Incomplete Understanding Unresponsiveness could also arise through incomplete conceptual understanding of the valuation task for a variety of reasons. For example, a subject may understand that G92 is more preferable than G8, but does not understand *by how much*. Another example is when a subject has no clear understanding of how to translate lottery primitives into monetary valuations.

Taking advantage of the calculator input data, I can identify a subset of calculations that are strongly indicative of incomplete understanding. This identification process uses the feature of procedures defined in Section 4.3, which capture the functional form that the subjects use to map the primitives of the task to their valuations. If a procedure is strictly decreasing in any of the monetary outcomes, the procedure would imply a lower valuation when the monetary outcome increases. Similarly, if a procedure is a strictly decreasing function in b_1, the procedure would imply a lower (absolute) valuation when the probability of the non-zero outcome increases. These two types of procedures, when observed, strongly suggest that the observed valuations are due to incomplete understanding. I refer to these two types of procedures as *decreasing procedures*.[17]

The identification of decreasing procedures provides stronger evidence of incomplete understanding than simply observing valuations that violate monotonicity. While monotonicity violations in stated valuations could arise from multiple sources – including typos, cognitive imprecision (Khaw, Li and Woodford, 2021), or other factors that introduce randomness

[17] Formally, I define a procedure to be decreasing if the procedure (after substituting b_2 with $100 - b_1$) is strictly decreasing in any of the primitives m_1, m_2, or b_1, at any point within the range $\{(m_1, m_2, b_1) : m_1 \geq 0, m_2 \geq 0, 0 \leq b_1 \leq 100\}$. For example, among all procedures listed in Panel A of Table 5, there are two decreasing procedures: $b_2 \times m_1/100$ and $100/b_1$.

into responses – such violations do not necessarily demonstrate that subjects fundamentally misunderstand some aspects of the lotteries or the valuation task. In contrast, decreasing procedures reveal systematic errors in subjects' approach to the task itself. When a subject applies a decreasing procedure, this reflects a deliberate approach that leads to incorrect valuations. The procedural data thus provides direct evidence of incomplete understanding: subjects are not merely making noisy responses around a correct understanding, but are systematically implementing flawed approaches to lottery valuation.

Subject Type	Lottery % Decreasing	Mirror % Decreasing
Expected value	2.2%	2.2%
Number	3.3%	3.3%
Linear money	5.0%	6.1%
Linear box	2.5%	16.7%
Nonlinear	39.9%	37.9%

Table 9: Share of decreasing procedures by subject type and task type

Table 9 shows the shares of decreasing procedures for each subject type. Nonlinear type subjects disproportionately use decreasing procedures in both lottery and mirror tasks, while other types mostly avoid using decreasing procedures. It is important to note that while the presence of a decreasing procedure strongly suggests the presence of incomplete understanding (sufficiency), the absence of a decreasing procedure does not meaningfully suggest the absence of incomplete understanding (necessity). For example, a number procedure is by definition never decreasing, since none of the primitives appears in the procedure.[18] But a number procedure may still be a result from incomplete understanding.

Result 7. *There is strong evidence supporting the presence of incomplete understanding in the non-linear type subjects, since these subjects disproportionately use decreasing procedures.*

6.3 What Can Explain the Linear Money Calculations?

Linear money calculations are the third-largest group of calculations, behind the easily-interpretable EV group and the largely uninterpretable number group. Thus, linear money

[18]Though number (and expected value) procedures are by definition not decreasing, since Table 9 shows the shares of decreasing procedures by *subject* type instead of *procedure* group, the shares of decreasing procedures among these subject types can still be positive.

calculations offer unique insights into the decision-making processes of an important group of subjects. Here, I examine whether two theories – probability weighting and attribute substitution – can explain the appearance of these calculations and their resulting valuations in the data.

Probability Weighting Probability weighting is when economic agents evaluate lotteries by applying subjective probability weights to monetary outcomes rather than using the objective probabilities. Formally, these weights are determined by a probability weighting function $w(\cdot)$ that transforms objective probabilities into probability weights. In the current context, assuming the utility is linear for small stakes, for a lottery Gn, the valuation generated by probability weighting would be $w(n/100) \times 26$, where $w(\cdot)$ is the probability weighting function, and analogously for a lottery Ln.

If we take probability weighting as a literal description of the valuation process, it has the potential to explain the appearance of linear money procedures. To illustrate, consider an agent valuing (G75, Lottery) by literally implementing probability weighting. This agent would calculate their valuation using the expression $w(0.75) \times 26$. If, for example, $w(0.75) = 0.6$, the agent would perform 0.6×26 in the calculator. From the experimenter's perspective, this calculation would appear as a linear money procedure – specifically, one that multiplies the monetary outcome (26) by a coefficient (0.6) that differs from the true probability (0.75) and cannot be matched with any task primitives.

Moreover, when the linear money procedures are interpreted as literally implementing probability weighting, it should be possible to directly recover subjects' probability weighting function from their calculations. Specifically, when a subject valuing a lottery Gn/Ln uses a linear money procedure that assigns coefficient γ_1 to the money amount m_1, this reveals that their probability weight would be $w(n/100) = \gamma_1$ under this interpretive framework. To invert the example in the previous paragraph, if I see a subject using the procedure $0.6 \times m_1$ when valuing (G75, Lottery), through the lens of probability weighting, I can infer that the probability weight is $w(0.75) = 0.6$.[19]

[19] I only use the coefficient of m_1, but not the coefficient of m_2, to recover the probability weighting function. This is because m_2 is always zero in our experimental design. As a result, subjects may rationally omit terms involving m_2 from their calculations, making it impossible to reliably recover the probability weights placed on m_2.

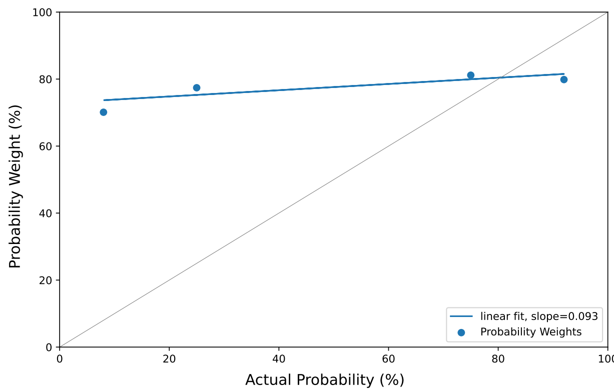


Figure 7: Probability weighting function recovered from linear money procedures for linear money subjects

Figure 7 displays the recovered probability weighting function as the average of the recovered probability weights. The sample is restricted to the rounds where linear money procedures are used by linear money-type subjects.[20] The extremely low slope (0.093) of the probability weighting function demonstrates that the probability weights are almost entirely unresponsive to changes in the actual probability. Such extreme unresponsiveness is difficult to reconcile with probability weighting as a theoretical account of these subjects' calculations. Moreover, the probability weighting function recovered using the calculations presents a stark contrast to the inverse S-shaped probability weighting function typically recovered using valuation data (e.g., Gonzalez and Wu, 1999). Thus, while subjects' use of linear money procedures superficially aligns with probability weighting's functional form predictions, evidence from the coefficients in these linear money procedures casts significant doubt on probability weighting as the underlying mechanism leading to these calculations.

It is important to note that this analysis should not be interpreted as rejecting probability weighting as a descriptive model of choice patterns. Rather, it suggests that when subjects engage in explicit valuation calculations, they do not appear to be literally implementing probability weighting procedures. The theory may still provide an accurate "as-if" characterization if, for instance, non-linear probability cognition automatically generates probability weighting patterns at a subconscious level.

Attribute Substitution The theory of attribute substitution (Kahneman and Frederick, 2002) provides a framework for understanding the linear money calculations observed in

[20]Expanding the sample to all linear money procedures regardless of subject type leads to similar results.

the data. According to this theory, "an individual assesses a specified target attribute of a judgment object by substituting another property of that object – the heuristic attribute – which comes more readily to mind" (p. 53). For attribute substitution to govern judgment, the necessary conditions include: "(1) the target attribute is relatively inaccessible; and (2) a semantically and associatively related attribute is highly accessible" (p. 54). In the context of lottery valuation tasks, these conditions are plausibly satisfied: the target attribute (the valuation according to the preference) may be relatively inaccessible, while the heuristic attribute (the monetary outcomes of the lottery) is highly accessible and semantically related to valuation since both involve monetary amounts.

The observed prevalence of linear money calculations in the data is consistent with attribute substitution theory, though the theory does not a priori predict this specific functional form. Rather, having documented these linear patterns empirically, one can retrospectively understand why they represent a natural manifestation of attribute substitution. When subjects substitute monetary outcomes for valuations, they must still transform these monetary amounts into their stated valuations through some functional relationship. Linear transformations emerge as a particularly natural choice given their simplicity.

This theory explains the extreme unresponsiveness to probabilities through a specific mechanism: when subjects substitute monetary outcomes for proper lottery valuations, they are essentially replacing a complex task (combining probabilities and outcomes to form their valuations) with a simpler one (focusing primarily on the monetary amounts). This substitution leads to a decision-making process that is primarily based on the monetary outcomes, and inherently neglects the probabilities.

The attribute substitution theory emerging from the calculator input data can also explain an empirical pattern observed across multiple studies: lottery valuations are much more responsive to changes in monetary outcomes than to changes in probabilities, particularly in the small-stakes experimental settings that are common in the literature. Attribute substitution theory provides a compelling explanation for this asymmetric responsiveness because subjects who substitute monetary outcomes for proper valuations attend primarily to the monetary amounts, while probabilities become largely irrelevant to their decision-making.

To illustrate this asymmetric responsiveness, I analyze lottery valuation data from Enke

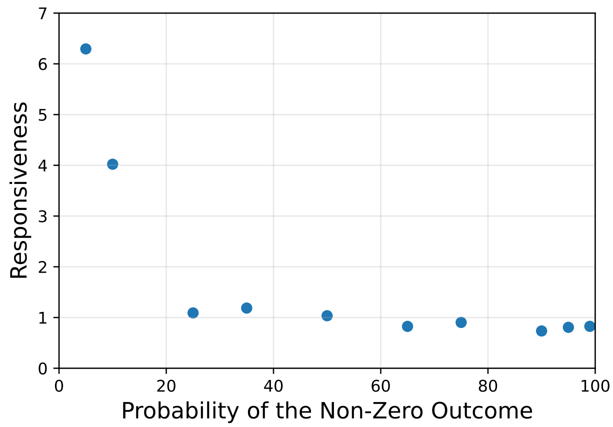


Figure 8: Responsiveness of valuations to the positive monetary outcome across different probabilities. The data is from Enke and Graeber (2023).

and Graeber (2023), which contains valuations of lotteries in the form of $(\$X, p; 0)$ that vary systematically along both probability (p) and monetary outcome ($X > 0$) dimensions. I measure responsiveness by regressing valuations on expected values within subsets of lotteries that hold one dimension constant while varying the other – this generates separate responsiveness measures for probability changes (using lotteries with identical monetary outcomes but different probabilities) and monetary outcome changes (using lotteries with identical probabilities but different monetary outcomes).[21] Across all probabilities of the non-zero outcome (5%-99%), responsiveness to monetary outcomes is remarkably high, ranging from 0.83 to 6.29 (see Figure 8). In contrast, responsiveness to probabilities is substantially lower across all non-zero monetary outcomes ($15-$25), ranging from 0.48 to 0.59. The asymmetric responsiveness has considerable theoretical significance – under standard expected utility theory with any strictly concave utility function, valuations should be *less* responsive to changes in the monetary outcome than to changes in probabilities – precisely the opposite of what is observed empirically.[22]

The most prevalent form of attribute substitution directly observed in the calculator input data – simply using the non-zero monetary outcome as the valuation (the most common linear money procedure, see Table 5) – offers an additional testable prediction: responsiveness

[21]Again, a risk neutral agent would exhibit unit responsiveness to monetary outcomes ($r = 1$).

[22]For any binary-outcome lottery $(X, p; Y)$ where $X > Y$, under expected utility with concave utility function, it is straightforward to show that responsiveness to X is greater than responsiveness to p, and in turn, greater than responsiveness to Y. In Enke and Graeber (2023), Y is fixed at $0, and the responsiveness of interest is to X (the non-zero monetary outcome).

to monetary outcomes should be higher when the probability of the non-zero monetary outcome is smaller. The logic is straightforward: for subjects employing this form of attribute substitution, increasing the non-zero monetary outcome by \$1 increases their valuation by \$1, but increases the expected value by only $\$1 \times p$ (where p is the probability of the non-zero outcome). Therefore, the responsiveness (change in valuation divided by change in expected value) equals $1/p$ – making responsiveness inversely related to probability. This prediction is verified in Figure 8, which shows that responsiveness to the non-zero monetary outcome decreases markedly as the probability increases. With small probabilities of the non-zero outcome, valuations are extremely responsive to changes in the non-zero outcome, with responsiveness reaching 6.29 at 5% probability and 4.02 at 10% probability. The responsiveness remains above 1 for probabilities of 25%, 35%, and 50%, and becomes smaller than 1 for probabilities of 65% and beyond, clearly demonstrating the predicted inverse relationship between probability and responsiveness.

Result 8. *The linear money type subjects are consistent with the attribute substitution theory. These subjects base their valuations predominantly on the monetary outcomes, while largely neglecting the probabilities.*

6.4 Summary

Combining evidence from the three perspectives, the calculator input data provide new insights into the theoretical mechanisms driving unresponsiveness. As a starting point, although some subjects may have primarily expressed their risk preferences when valuing lotteries in the NoCalc treatment, most subjects' behavior is inconsistent with models that assume valuations solely reflect risk preferences. Separate examination of each subject type reveals distinct mechanisms driving the unresponsiveness of each type. First, the unresponsiveness exhibited by EV-type subjects is primarily consistent with implementation costs. Second, for number-type subjects, the underlying mechanism is less clear given the limited interpretability of their calculations. Third, the unresponsiveness of linear money-type subjects provides strong evidence for the attribute substitution theory. Finally, there is strong evidence that many non-linear type subjects suffer from incomplete understanding.

7 Connections to the Literature

This study makes contributions to a few strands of literature. First, it contributes to the literature that studies the roles played by cognitive complexity in the measured risk attitudes. The literature has amassed significant evidence that, at least to some extent, the observed departure of small-stake lottery valuations from their expected values are consequences of cognitive complexity, as opposed to fully reflecting the risk preferences (Harbaugh, Krause and Vesterlund, 2010, Woodford, 2012, Benjamin, Brown and Shapiro, 2013, Martínez-Marquina, Niederle and Vespa, 2019, Khaw, Li and Woodford, 2021, Frydman and Jin, 2021, Nielsen and Rehbeck, 2022, Choi et al., 2022, Enke and Graeber, 2023, Enke and Shubatt, 2023, Oprea, 2024*b*, Puri, 2025). As to the exact nature of this complexity, the extant literature has proposed a few candidate sources, but has not yet reached a consensus over the most important sources of complexity. My main contribution to this literature is that, by revealing subjects' calculations behind the elicited lottery valuations and linking the calculations to measured risk attitudes, I provide new insights on the sources of this complexity.

Second, the study contributes to the interdisciplinary literature that measures the decision-making processes that generate observed choices. To reveal this usually unobserved layer, studies in this literature use various techniques including mouse tracking (Payne, Bettman and Johnson, 1988), eye tracking (Reutskaja et al., 2011), intermediate choice tracking (Caplin, Dean and Martin, 2011), and verbal descriptions of decision-making processes (Ericsson and Simon, 1980, Arrieta and Nielsen, 2023). Most relevantly, a branch of this literature has applied these techniques to risk attitudes, the very question studied here (Payne, Bettman and Johnson, 1988, Arieli, Ben-Ami and Rubinstein, 2011, Pachur et al., 2013, 2018, Arrieta and Nielsen, 2023). The techniques developed so far have mostly focused on the information acquisition aspect of the decision-making process, that is, what information is accessed. This study makes an important methodological contribution to this literature by developing the calculator design that recovers the computational aspect of the decision-making process, i.e., how decision-makers utilize the accessed information to perform calculations, which cannot be recovered by previous techniques. The calculator design developed in this study, along with the features of procedures and base terms, can be easily transplanted and deployed to studying the computational aspect of the decision-making processes in other domains of individual and strategic decision-making.

This study also contributes to the literature that explicitly models and measures procedural decision-making (Simon, 1955, Payne, Bettman and Johnson, 1988, Oprea, 2020, Arrieta and Nielsen, 2023, Banovetz and Oprea, 2023, Oprea, 2024*a*). This literature takes procedures as the fundamental object that economic decision-makers need to choose in the decision-making processes. While the standard theory aims to describe how people choose from feasible *actions*, this literature aims to describe how people choose from feasible *procedures*, each of which is a mapping from the task primitives to the actions. A particular focus of this literature is how the characteristics of the decision-making environment and the implementation costs of these procedures affect the use of these procedures and the actions resulting from these procedures. The current study records the computational aspect of the procedures and measures the implementation costs, and thus provides direct tests of the predictions made by this literature.

More broadly, this study joins a long list of literature studying anomalies in choices under risk (e.g., Kahneman and Tversky, 1979, Tversky and Kahneman, 1992, Gonzalez and Wu, 1999, Wakker, 2010, Bruhin, Fehr-Duda and Epper, 2010, O'Donoghue and Somerville, 2018, Beauchamp et al., 2020, Bernheim and Sprenger, 2020, Oprea, 2024*b*, McGranaghan et al., 2024). This study micro-founds the observed unresponsiveness and fourfold patterns in lottery valuations by documenting the decision-making processes behind these valuations.

References

Arieli, Amos, Yaniv Ben-Ami, and Ariel Rubinstein. 2011. "Tracking Decision Makers under Uncertainty." *American Economic Journal: Microeconomics*, 3(4): 68–76.

Arrieta, Gonzalo, and Kirby Nielsen. 2023. "Procedural Decision-Making In The Face Of Complexity."

Banki, Daniel, Uri Simonsohn, Robert Walatka, and George Wu. 2025. "Decisions under Risk Are Decisions under Complexity: Comment."

Banovetz, James, and Ryan Oprea. 2023. "Complexity and Procedural Choice." *American Economic Journal: Microeconomics*, 15(2): 384–413.

Beauchamp, Jonathan P., Daniel J. Benjamin, David I. Laibson, and Christopher F. Chabris. 2020. "Measuring and Controlling for the Compromise Effect When Estimating Risk Preference Parameters." *Experimental Economics*, 23(4): 1069–1099.

Becker, Gordon M., Morris H. Degroot, and Jacob Marschak. 1964. "Measuring Utility by a Single-Response Sequential Method." *Behavioral Science*, 9(3): 226–232.

Benjamin, Daniel J., Sebastian A. Brown, and Jesse M. Shapiro. 2013. "Who Is 'Behavioral'? Cognitive Ability and Anomalous Preferences." *Journal of the European Economic Association*, 11(6): 1231–1255.

Bernheim, B. Douglas, and Charles Sprenger. 2020. "On the Empirical Validity of Cumulative Prospect Theory: Experimental Evidence of Rank-Independent Probability Weighting." *Econometrica*, 88(4): 1363–1409.

Blavatskyy, Pavlo R. 2007. "Stochastic Expected Utility Theory." *Journal of Risk and Uncertainty*, 34(3): 259–286.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. "Latent Dirichlet Allocation." *J. Mach. Learn. Res.*, 3(null): 993–1022.

Bruhin, Adrian, Helga Fehr-Duda, and Thomas Epper. 2010. "Risk and Rationality: Uncovering Heterogeneity in Probability Distortion." *Econometrica*, 78(4): 1375–1412.

Caplin, Andrew, Mark Dean, and Daniel Martin. 2011. "Search and Satisficing." *American Economic Review*, 101(7): 2899–2922.

Chen, Daniel L., Martin Schonger, and Chris Wickens. 2016. "oTree—An Open-Source Platform for Laboratory, Online, and Field Experiments." *Journal of Behavioral and Experimental Finance*, 9: 88–97.

Choi, Syngjoo, Jeongbin Kim, Eungik Lee, and Jungmin Lee. 2022. "Probability Weighting and Cognitive Ability." *Management Science*.

Enke, Benjamin, and Cassidy Shubatt. 2023. "Quantifying Lottery Choice Complexity." National Bureau of Economic Research w31677, Cambridge, MA.

Enke, Benjamin, and Thomas Graeber. 2023. "Cognitive Uncertainty*." *The Quarterly Journal of Economics*, 138(4): 2021–2067.

Enke, Benjamin, Thomas Graeber, Ryan Oprea, and Jeffrey Yang. 2024. "Behavioral Attenuation." National Bureau of Economic Research w32973, Cambridge, MA.

Enke, Benjamin, Uri Gneezy, Brian Hall, David Martin, Vadim Nelidov, Theo Offerman, and Jeroen van de Ven. 2023. "Cognitive Biases: Mistakes or Missing Stakes?" *The Review of Economics and Statistics*, 105(4): 818–832.

Ericsson, K. Anders, and Herbert A. Simon. 1980. "Verbal Reports as Data." *Psychological Review*, 87(3): 215–251.

Frydman, Cary, and Lawrence J Jin. 2021. "Efficient Coding and Risky Choice." *The Quarterly Journal of Economics*, 137(1): 161–213.

Gao, Yihong, and Michele Garagnani. 2025. "Improving Risky Choices: The Effect of Cognitive Offloading on Risky Decisions." *Journal of Risk and Uncertainty*, 70(2): 105–128.

Gonzalez, Richard, and George Wu. 1999. "On the Shape of the Probability Weighting Function." *Cognitive Psychology*, 38(1): 129–166.

Handel, Benjamin, and Joshua Schwartzstein. 2018. "Frictions or Mental Gaps: What's Behind the Information We (Don't) Use and When Do We Care?" *Journal of Economic Perspectives*, 32(1): 155–178.

Harbaugh, William T, Kate Krause, and Lise Vesterlund. 2010. "The Fourfold Pattern of Risk Attitudes in Choice and Pricing Tasks." *The Economic Journal*, 120(545): 595–611.

Healy, Paul J. 2020. "Explaining the BDM—or Any Random Binary Choice Elicitation Mechanism—to Subjects."

Kahneman, Daniel, and Amos Tversky. 1979. "Prospect Theory: An Analysis of Decision under Risk." *Econometrica*, 47(2): 263–291.

Kahneman, Daniel, and Shane Frederick. 2002. "Representativeness Revisited: Attribute Substitution in Intuitive Judgment." In *Heuristics and Biases: The Psychology of Intuitive Judgment*. 49–81. New York, NY, US:Cambridge University Press.

Khaw, Mel Win, Ziang Li, and Michael Woodford. 2021. "Cognitive Imprecision and Small-Stakes Risk Aversion." *The Review of Economic Studies*, 88(4): 1979–2013.

l'Haridon, Olivier, and Ferdinand M. Vieider. 2019. "All over the Map: A Worldwide Comparison of Risk Preferences." *Quantitative Economics*, 10(1): 185–215.

Li, Jianbiao, Zenghui Liu, Xiaofei Niu, and Yue Wang. 2025. "Decisions Under Risk Are Not Decisions Under Complexity."

Martínez-Marquina, Alejandro, Muriel Niederle, and Emanuel Vespa. 2019. "Failures in Contingent Reasoning: The Role of Uncertainty." *American Economic Review*, 109(10): 3437–3474.

McGranaghan, Christina, Kirby Nielsen, Ted O'Donoghue, Jason Somerville, and Charles D. Sprenger. 2024. "Distinguishing Common Ratio Preferences from Common Ratio Effects Using Paired Valuation Tasks." *American Economic Review*, 114(2): 307–347.

Nielsen, Kirby, and John Rehbeck. 2022. "When Choices Are Mistakes." *American Economic Review*, 112(7): 2237–2268.

O'Donoghue, Ted, and Jason Somerville. 2018. "Modeling Risk Aversion in Economics." *Journal of Economic Perspectives*, 32(2): 91–114.

Oprea, Ryan. 2020. "What Makes a Rule Complex?" *American Economic Review*, 110(12): 3913–3951.

Oprea, Ryan. 2024*a*. "Complexity and Its Measurement."

Oprea, Ryan. 2024*b*. "Decisions under Risk Are Decisions under Complexity." *American Economic Review*, 114(12): 3789–3811.

Pachur, Thorsten, Michael Schulte-Mecklenbeck, Ryan O Murphy, and Ralph Hertwig. 2018. "Prospect Theory Reflects Selective Allocation of Attention." *Journal of Experimental Psychology: General*.

Pachur, Thorsten, Ralph Hertwig, Gerd Gigerenzer, and Eduard Brandstätter. 2013. "Testing Process Predictions of Models of Risky Choice: A Quantitative Model Comparison Approach." *Frontiers in Psychology*, 4.

Payne, John W., James R. Bettman, and Eric J. Johnson. 1988. "Adaptive Strategy Selection in Decision Making." *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14(3): 534–552.

Puri, Indira. 2025. "Simplicity and Risk." *The Journal of Finance*, 80(2): 1029–1080.

Reutskaja, Elena, Rosemarie Nagel, Colin F. Camerer, and Antonio Rangel. 2011. "Search Dynamics in Consumer Choice under Time Pressure: An Eye-Tracking Study." *American Economic Review*, 101(2): 900–926.

Simon, Herbert A. 1955. "A Behavioral Model of Rational Choice." *The Quarterly Journal of Economics*, 69(1): 99–118.

Tversky, Amos, and Daniel Kahneman. 1992. "Advances in Prospect Theory: Cumulative Representation of Uncertainty." *Journal of Risk and Uncertainty*, 5(4): 297–323.

Wakker, Peter P. 2010. *Prospect Theory: For Risk and Ambiguity.* Cambridge:Cambridge University Press.

Wakker, Peter P. 2025. "Relating Risky to Riskless Preferences, and Their Joint Irrationality: A Comment on Oprea (2024)."

Woodford, Michael. 2012. "Inattentive Valuation and Reference-Dependent Choice."

Appendices

A Additional Figures and Tables

	lottery procedure group				
mirror procedure group	expected value	linear box	linear money	nonlinear	number
expected value	28.9%	0.2%	2.9%	2.5%	2.5%
linear box	0.7%	1.6%	0.5%	0.3%	1.0%
linear money	1.8%	0.5%	9.0%	0.8%	2.7%
nonlinear	2.2%	0.6%	1.4%	4.5%	0.8%
number	1.1%	0.9%	6.9%	0.9%	24.9%
P(same group|lottery group)	83.4%	42.6%	43.5%	50.0%	78.1%

Table A.1: Joint distribution of lottery procedure groups and mirror procedure groups, among within-subject pairs. The final row is the probability that the lottery and mirror procedure groups are the same, conditional on the lottery procedure group.

	Subject Type				
Base Term Group	expected value	number	linear money	linear box	nonlinear
expected value	**88.0%**	12.8%	16.3%	38.9%	23.8%
number	5.3%	**79.9%**	31.4%	6.7%	12.7%
linear money	11.7%	13.7%	**54.6%**	20.0%	25.4%
linear box	2.3%	5.7%	5.6%	**61.1%**	9.5%
nonlinear	7.2%	2.9%	3.9%	7.8%	**63.5%**

Table A.2: Fractions of mirror rounds where each base term group (the row) appear, conditional on subject type (the column)

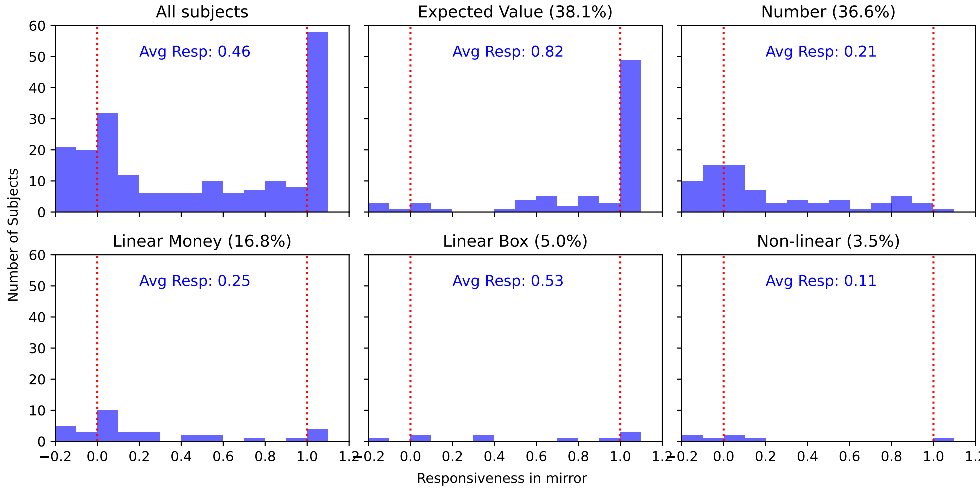


Figure A.1: Histograms of individualnresponsiveness in mirror tasks in the Calc treatment

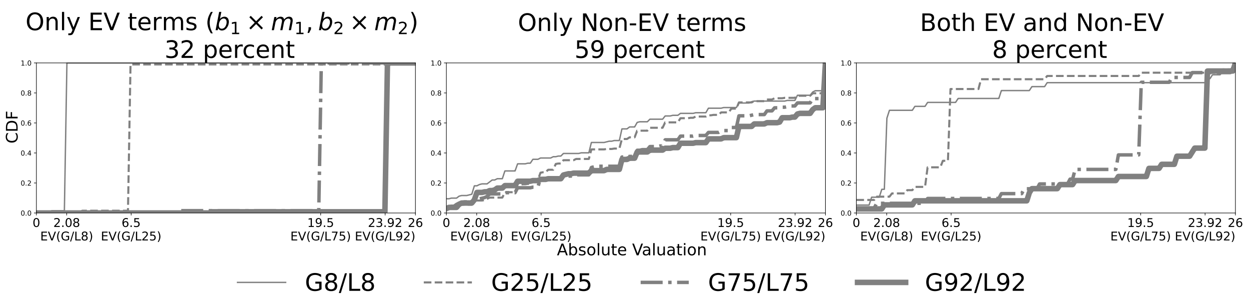


Figure A.2: CDF of lottery valuations for three disjoint groups of rounds

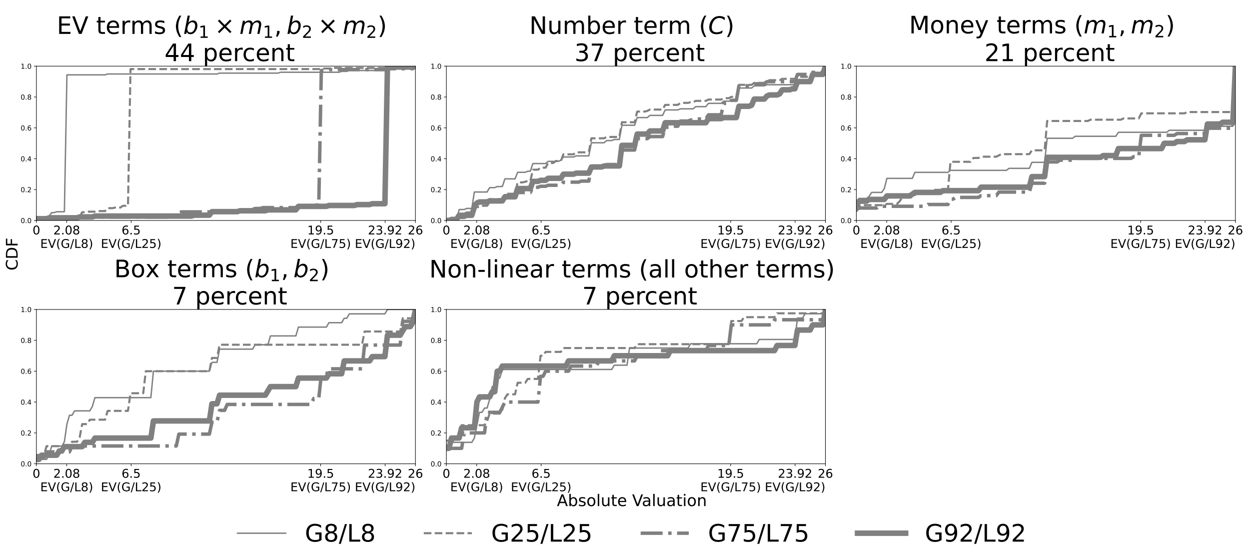


Figure A.3: CDF of mirror valuations in the NoCalc treatment, conditional on employing each group of base terms.

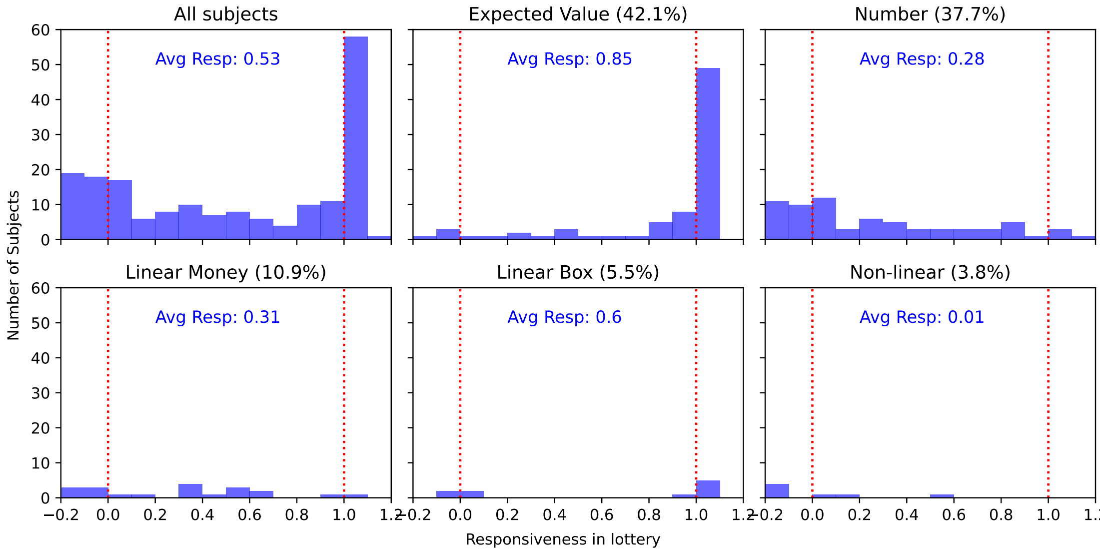


Figure A.4: Histograms of individual responsiveness in lottery tasks in the Calc treatment, excluding those subjects who give the same valuation in no fewer than 7 out of a total of 9 tasks, and separately for each type of subjects.

To examine the external validity of Result 3, for each round in the NoCalc treatment, I match it with the round in the Calc treatment where the same subject faces the same lottery task. I then reproduce Figure 4 using valuations from the NoCalc rounds and the calculator input from their matched Calc rounds. The reproduced graph can be found in Appendix Figure A.5. The headline observations from Result 3 still hold. NoCalc rounds where the subjects employ an EV term in their matched Calc rounds generate responsive and concentrated lottery valuations, though they are not quite to the same degree as their matched Calc rounds. Moreover, NoCalc rounds where the subjects employ any non-EV term in their matched Calc rounds again generate unresponsive and dispersed lottery valuations. Many of the idiosyncratic patterns conditional on groups are also preserved. For example, conditional on the matched Calc round using a money base term, the NoCalc valuations again have large point masses at \$26 (the maximum absolute valuation).

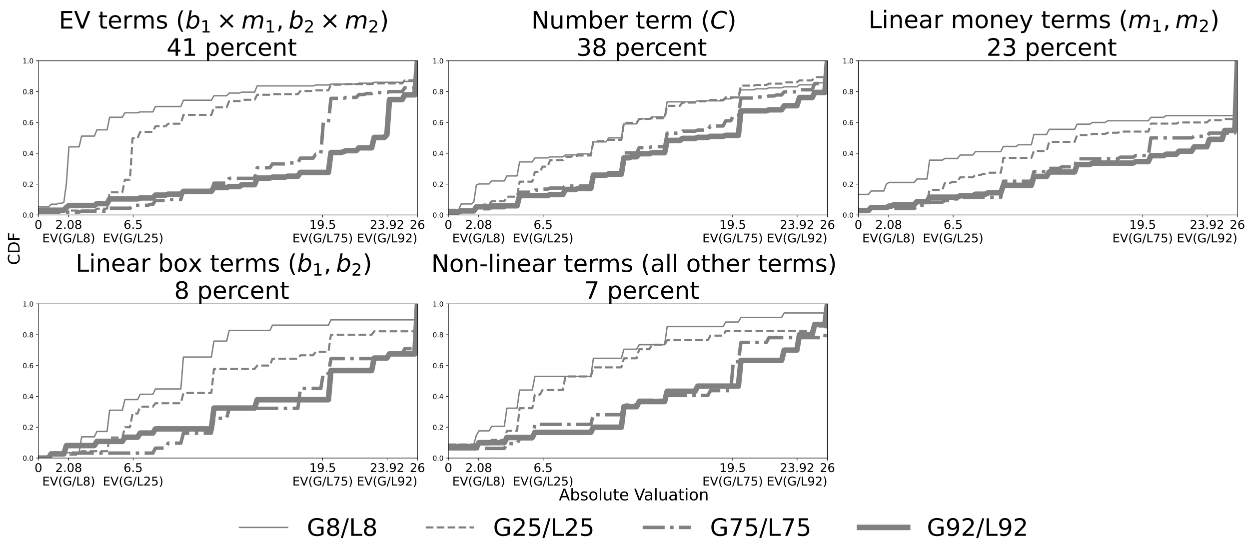


Figure A.5: Distributions of lottery valuations in the NoCalc treatment, grouped by the base terms used by the same subjects in the corresponding Calc treatment rounds (see the text for details)

B Topic Modeling of Calculator Inputs

Latent Dirichlet Allocation (LDA) is an unsupervised machine learning technique designed to discover hidden thematic structures in collections of documents (Blei, Ng and Jordan, 2003). In the context of this paper, I apply LDA to uncover natural groupings in the calculator inputs provided by subjects.

LDA operates on the intuition that documents (in this case, calculator inputs) can be represented as mixtures of *topics*, where each topic is characterized by a distribution over words (in this case, base terms). LDA simultaneously estimates both the topic composition of each document and the word distribution of each topic. Formally, LDA models each topic $k \in \{1, 2, \ldots, K\}$ as a multinomial distribution $\boldsymbol{\phi}_k$ over the vocabulary of base terms. Each $\boldsymbol{\phi}_k$ is a vector where the component $\phi_{k,v}$ represents the probability of base term v appearing in topic k. These probabilities satisfy $\sum_v \phi_{k,v} = 1$ for all topics k. For each calculator input d, LDA also estimates a topic mixture $\boldsymbol{\theta}_d$, where each component $\theta_{d,k}$ represents the proportion of terms in document d that are drawn from topic k.

The key advantage of using LDA in this study is that it allows for the identification of "semantic" relationship between base terms without imposing a predetermined structure. Rather than manually categorizing base terms, LDA provides an unsupervised, data-driven approach to uncovering natural groupings based on how base terms co-occur within calculator inputs. This helps validate the intuitive categorization of base terms into the five groups (Expected value, Number, Linear money, Linear box, and Non-linear) used in the main analysis.

To implement the LDA model, I represented each calculator input in lottery tasks as a "document" defined by its corresponding base term set. This approach treats the collection of base terms associated with a calculator input as analogous to the words in a text document. For model specification, I set the number of topics (a hyperparameter that needs to be manually set) $K = 5$ to align with the five base term groups hypothesized in the main text. This parameter choice facilitates direct comparison between the data-driven topics and the conceptually defined groups.

Appendix Figure B.1 presents the LDA-generated topics and the associated probabilities of base terms appearing in each topic. The clustering of specific base terms within topics reflects their tendency to co-occur in subjects' calculations, providing a natural basis for grouping functionally similar terms. The results strongly support the classification of base terms into the five groups used in the main analysis. Topic 0 (34.1%) is dominated by terms involving the product of box quantities and monetary amounts ($b_1 \times m_1, b_2 \times m_2$), clearly corresponding to the expected value group. Topic 1 (33.4%) is primarily characterized by the constant term C , validating the number group. Topic 2 (19.5%) shows high probabilities for the monetary terms m_1 and m_2, aligning with the linear money group. Topic 3 (9.9%) features box quantities b_1 and b_2 as its most prominent terms, corresponding to the linear box group. Finally, Topic 4 (3.1%) captures various non-linear combinations of primitives ($b_2 \times m_1$, $1/b_1$, etc.) that do not fit into the other categories, supporting the non-linear group. This unsupervised classification thus provides strong empirical validation for the five base term groups employed throughout the paper.

It is worth noting that LDA estimation involves random initialization, which can affect the resulting topic and word distributions. Different initializations may produce somewhat different topic structures. However, in testing with multiple random initializations, I found that the vast majority of estimation runs generated at least 3-4 topics that clearly corresponded to the base term groups described above.

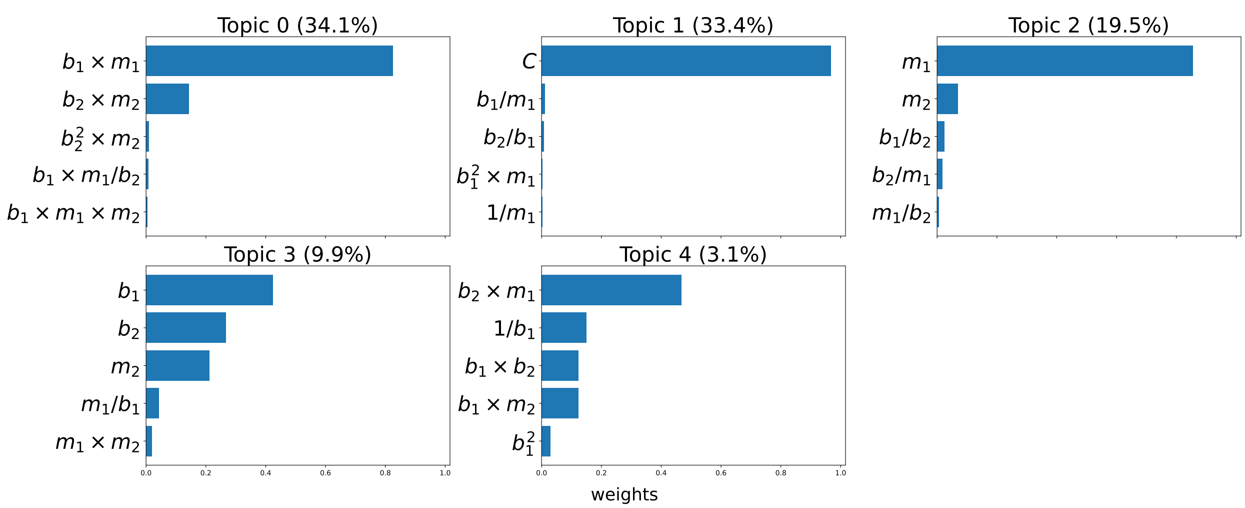


Figure B.1: LDA-generated topics and the probabilities of base terms in each topic

The topic model results can also be leveraged to construct an alternative approach to subject categorization. Rather than using modal base terms as described in Section 5.2, I aggregated the topic distributions of all calculator inputs to the subject level by averaging the document-topic mixtures ($\boldsymbol{\theta}_d$) across all documents (calculator inputs) produced by each subject in lottery tasks. This produces a subject-level topic distribution. I then classified each subject according to their dominant topic – the topic with the highest average probability. For example, subjects whose calculator inputs showed the highest average probability for Topic 0 (expected value) were classified as expected value type subjects. Remarkably, this LDA-based classification method yielded 95.0% agreement with the modal base term approach used in the main text. This high level of consistency between two methodologically distinct approaches to subject categorization provides strong evidence for the robustness of the subject type classifications and further validates the analysis of type-specific behaviors presented in the paper.

C Example Calculator Inputs

Below, I show randomly drawn example calculator inputs corresponding to the top 15 most frequent base term sets in lottery rounds. For each base term set, I give two examples, first a lottery round and then a mirror round. These top 15 base term sets collectively account for 90.5% of lottery rounds and 88.1% of mirror rounds.

Please note that, the experimental program elicits the willingness to *pay* for the lottery/mirror in gain tasks. However, the program instead elicits willingness to *prevent* (the maximum amount willing to forgo to prevent the lottery/mirror from happening) in loss tasks. In these tasks, the certainty equivalent is the negative of the willingness to prevent. As a result, the last row of the *Result* column represents the valuation in gains tasks, but instead the negative of the valuation in loss tasks. See Appendix G for the complete instructions.

- Base term set: $\{C\}$, frequency in lottery: 28.82%, frequency in mirror: 30.58%
  - Subject ID: 5iu9eyzk
  - Task: L25 lottery

Line	Numerical Expression	Result
1	24	24

  - Subject ID: 6n65prfw
  - Task: L92 mirror

Line	Numerical Expression	Result
1	7×3	21

- Base term set: $\{b_1 \times m_1\}$, frequency in lottery: 27.39%, frequency in mirror: 29.92%
  - Subject ID: 88171s3p
  - Task: L8 lottery

Line	Numerical Expression	Result
1	$8/100 \times 26$	2.08

- Subject ID: 7vzs5is7
- Task: L25 mirror

Line	Numerical Expression	Result
1	26×25	650
2	$650/100$	6.5

- Base term set: $\{m_1\}$, frequency in lottery: 13.20%, frequency in mirror: 10.29%
  - Subject ID: hgqqa7bl
  - Task: L75 lottery

Line	Numerical Expression	Result
1	26	26

  - Subject ID: id6e5867
  - Task: L92 mirror

Line	Numerical Expression	Result
1	26×1	26

- Base term set: $\{b_1 \times m_1, b_2 \times m_2\}$, frequency in lottery: 4.95%, frequency in mirror: 5.50%
  - Subject ID: cef3c6ir
  - Task: L25 lottery

Line	Numerical Expression	Result
1	$25 \times 26 + 75 \times 0/100$	650
2	$650/100$	6.5

  - Subject ID: cef3c6ir
  - Task: G92 mirror

Line	Numerical Expression	Result
1	$92 \times 26 + 8 \times 0$	2392
2	$2392/100$	23.92

- Base term set: $\{C, b_1 \times m_1\}$, frequency in lottery: 3.03%, frequency in mirror: 2.31%
  - Subject ID: qqoqux4t
  - Task: G25 lottery

Line	Numerical Expression	Result
1	$26 \times .25$	6.5
2	13	13

  - Subject ID: sysy515p
  - Task: G25 mirror

Line	Numerical Expression	Result
1	25×26	650
2	$Ans1/100$	6.5
3	$Ans2 - 3$	3.5

Please note that the calculator has an Ans button, which is a shortcut to use the result from the previous calculation. Ans1 refers to the result of Line 1, and similarly for others.

- Base term set: $\{C, m_1\}$, frequency in lottery: 2.75%, frequency in mirror: 1.60%
  - Subject ID: wshvt3vq
  - Task: G75 lottery

Line	Numerical Expression	Result
1	$26/2 + 1$	14

  - Subject ID: 668q3qjp

- Task: G25 mirror

Line	Numerical Expression	Result
1	$26/2$	13
2	$Ans1 - 8$	5

- Base term set: $\{m_2\}$, frequency in lottery: 1.98%, frequency in mirror: 1.21%
  - Subject ID: lhxiwlwe
  - Task: L8 lottery

Line	Numerical Expression	Result
1	2×0	0

  - Subject ID: l5hdvvcc
  - Task: L8 mirror

Line	Numerical Expression	Result
1	0	0

- Base term set: $\{b_1\}$, frequency in lottery: 1.38%, frequency in mirror: 1.82%
  - Subject ID: 9ny75gkz
  - Task: G25 lottery

Line	Numerical Expression	Result
1	30×0.25	7.5

  - Subject ID: y1vfs7ou
  - Task: G25 mirror

Line	Numerical Expression	Result
1	$30 \times .25$	7.5

- Base term set: $\{m_1, m_2\}$, frequency in lottery: 1.32%, frequency in mirror: 1.32%

  - Subject ID: 7k9gkzry
  - Task: L25 lottery

Line	Numerical Expression	Result
1	0.	0
2	26.00	26

  - Subject ID: pgwtf1kz
  - Task: G75 mirror

Line	Numerical Expression	Result
1	$26 + 0$	26

- Base term set: $\{b_1, b_1 \times m_1, b_2, b_2 \times m_2\}$, frequency in lottery: 1.27%, frequency in mirror: 0.00%
  - Subject ID: si04ncfg
  - Task: L25 lottery

Line	Numerical Expression	Result
1	$25/100$	0.25
2	$75/100$	0.75
3	$0.25 \times (26) + 0.75 \times 0$	6.5

  - This base term set does not appear in any mirror task

- Base term set: $\{b_1, b_1 \times m_1\}$, frequency in lottery: 1.16%, frequency in mirror: 0.94%
  - Subject ID: 7bfat155
  - Task: G25 lottery

Line	Numerical Expression	Result
1	$25/100$	0.25
2	0.25×26	6.5

- Subject ID: n4kcm1vv
- Task: G75 mirror

Line	Numerical Expression	Result
1	75×261	19575
2	75×26	1950
3	$1950/100$	19.5

- Base term set: $\{b_2 \times m_1\}$, frequency in lottery: 0.88%, frequency in mirror: 0.44%
  - Subject ID: 98qa7bdb
  - Task: G92 lottery

Line	Numerical Expression	Result
1	$.08 \times 26$	2.08

  - Subject ID: ze63xdkx
  - Task: L75 mirror

Line	Numerical Expression	Result
1	26×0.25	6.5

- Base term set: $\{C, m_2\}$, frequency in lottery: 0.83%, frequency in mirror: 0.33%
  - Subject ID: 00f2schi
  - Task: G25 lottery

Line	Numerical Expression	Result
1	$5 + 0$	5

  - Subject ID: ehfz4yq0
  - Task: L92 mirror

Line	Numerical Expression	Result
1	$0 + 6$	6

- Base term set: $\{b_1, b_2\}$, frequency in lottery: 0.77%, frequency in mirror: 0.94%
  - Subject ID: dwiup06z
  - Task: G25 lottery

Line	Numerical Expression	Result
1	$75 + 25$	100
2	$100/2$	50
3	$50/4$	12.5

  - Subject ID: dwiup06z
  - Task: L8 mirror

Line	Numerical Expression	Result
1	$92 + 8$	100
2	$100/2$	50
3	$50/4$	12.5

- Base term set: $\{b_2\}$, frequency in lottery: 0.77%, frequency in mirror: 0.88%
  - Subject ID: i1uphqli
  - Task: L25 lottery

Line	Numerical Expression	Result
1	$75/100$	0.75
2	0.75×30	22.5

  - Subject ID: i1uphqli
  - Task: G25 mirror

Line	Numerical Expression	Result
1	$75/100$	0.75
2	0.75×30	22.5

D Robustness of the Results to Understandings of Instructions

D.1 Replication with Instructions from Wu (2025) and Healy (2020)

The replication was conducted on Prolific in July 2025. A total of 49 subjects completed the replication. Each subject was paid a participation fee of \$7 for completing the experiment. With a 20% chance, a subject was also paid the outcome of a randomly chosen task. The median subject spent around 50 minutes on the experiment, and the average total earning from the experiment was \$11.23. The instructions for the replication can be found in Appendix H.

Appendix Figure D.1 replicates Figure 3. Out of the four observations in Section 3, two still hold in the replication: (1) unresponsiveness to changes in probabilities in lottery tasks; and (2) calculator increasing the responsiveness. However, two of the observations are weakened. First, as **?** points out, the responsiveness is now higher in mirrors compared with lotteries after comprehensive subject training and screening, but in the meantime, the responsiveness in mirror tasks is still far from the complete responsiveness benchmark (1). Second, gain and loss tasks are no longer symmetric, and gain tasks are significantly more responsive than loss tasks.

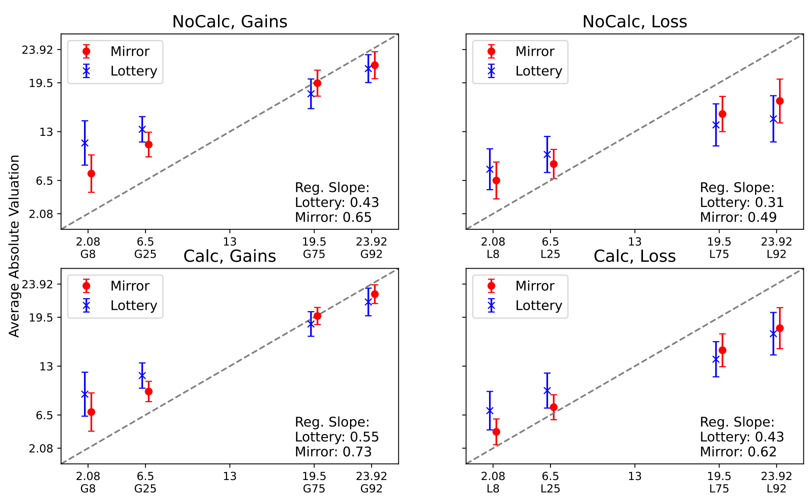


Figure D.1: Average valuation of each lottery and their deterministic mirror in the replication

Appendix Figure D.2 replicates Figure 4. The two salient observations in the main text regarding the discriminability and the dispersion of valuations conditional on employing a group of base terms still hold.

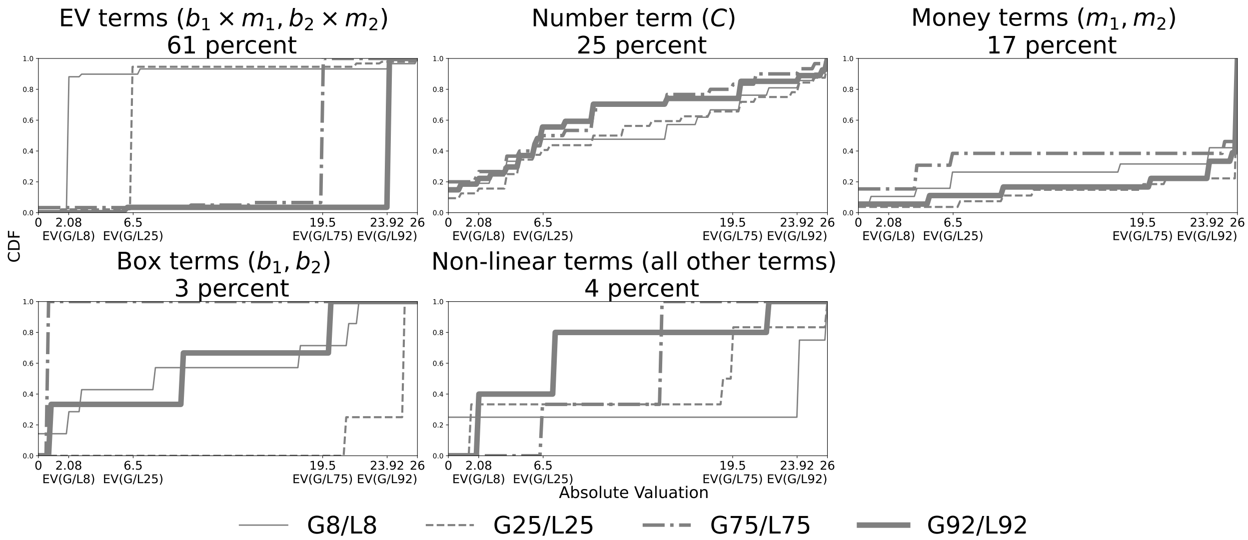


Figure D.2: Distributions of lottery valuations in the Calc treatment of the replication, conditional on employing each group of base terms.

Appendix Figure D.3 replicates Figure 5. The fraction of EV-types in the replication is higher than that in the full sample. The EV-type subjects are still close to complete responsiveness, but not as responsive as in the main experiment (average responsiveness = 0.78). The non-EV-type subjects (average responsiveness = -0.02) exhibit even more extreme unresponsiveness than in the main experiment (0.25). The unresponsiveness is so extreme that the non-EV-types violates monotonicity on average. The bimodal distribution of responsiveness is still present.

Appendix Figure D.4 replicates Figure 6. Again, the EV-types (average responsiveness = 0.48) are more responsive than the non-EV-types (0.07) in the NoCalc treatment. Comparing between treatments by comparing Appendix Figure D.2 and Appendix Figure D.3, the EV-types again show significantly higher responsiveness in the Calc treatment.

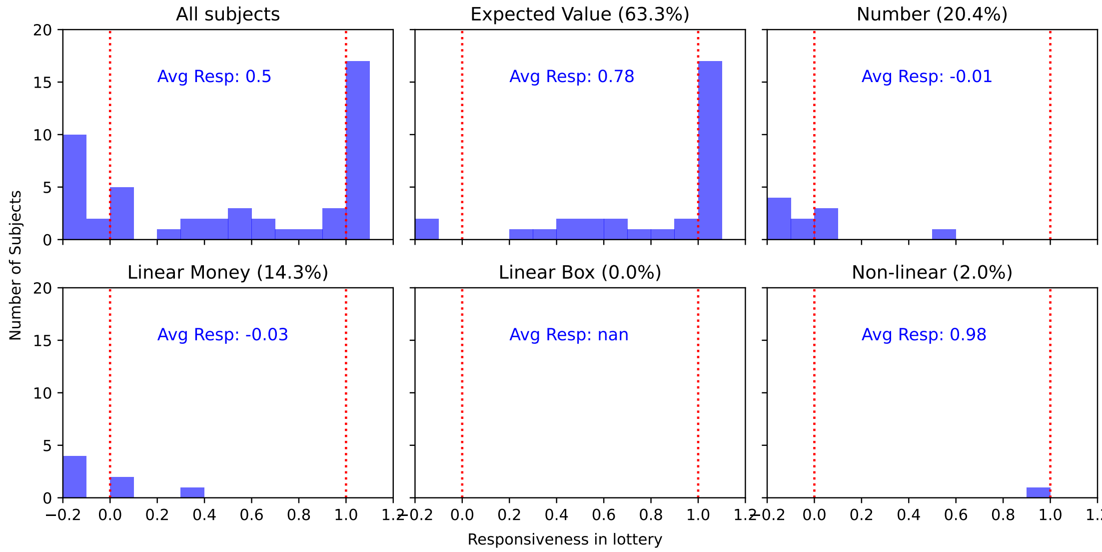


Figure D.3: Histograms of individual responsiveness in lottery tasks in the Calc treatment

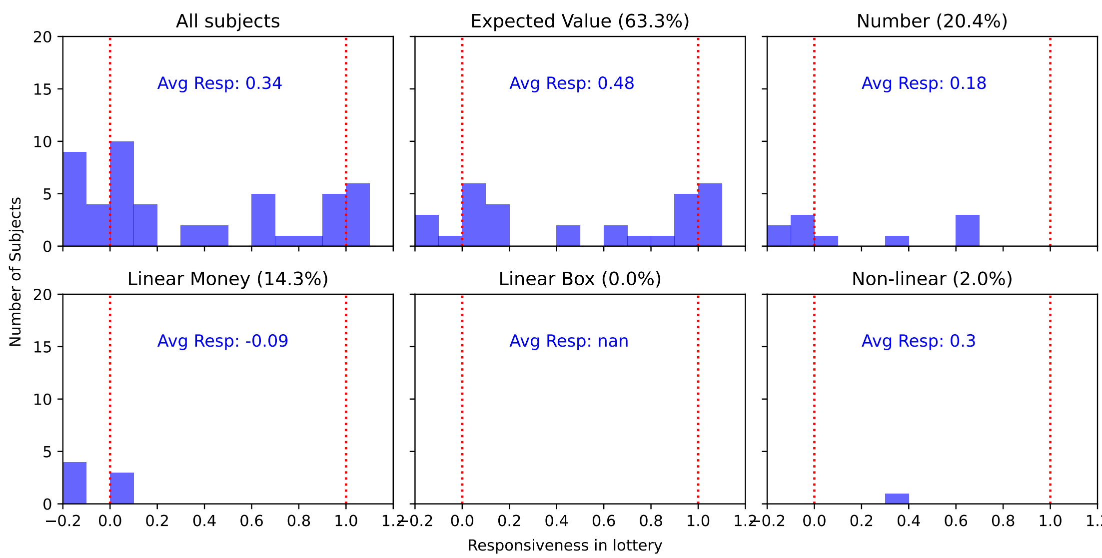


Figure D.4: Histograms of individual responsiveness in lottery tasks in the NoCalc treatment

D.2 Restricting the Sample as Proposed by Banki et al. (2025)

This appendix replicates the main results using only the 49 subjects who answer all four batches of comprehension questions correctly at their first trial. This sample selection is proposed by Banki et al. (2025) to address the potential confusion arising from Oprea's (2024*b*), and by extension, my experimental instructions. These subjects who correctly answer the comprehension questions are argued by Banki et al. (2025) to have better understanding of the experimental instructions. Especially, 20.4% of the subjects in this subset exhibit responsiveness smaller than 0.1 in the NoCalc treatment, which is more comparable to (but still higher than) the figures reported by most papers in the previous literature (see the end of Section 5.2).

Appendix Figure D.5 replicates Figure 3. The four observations in Section 3 still hold in this subsample of subjects.

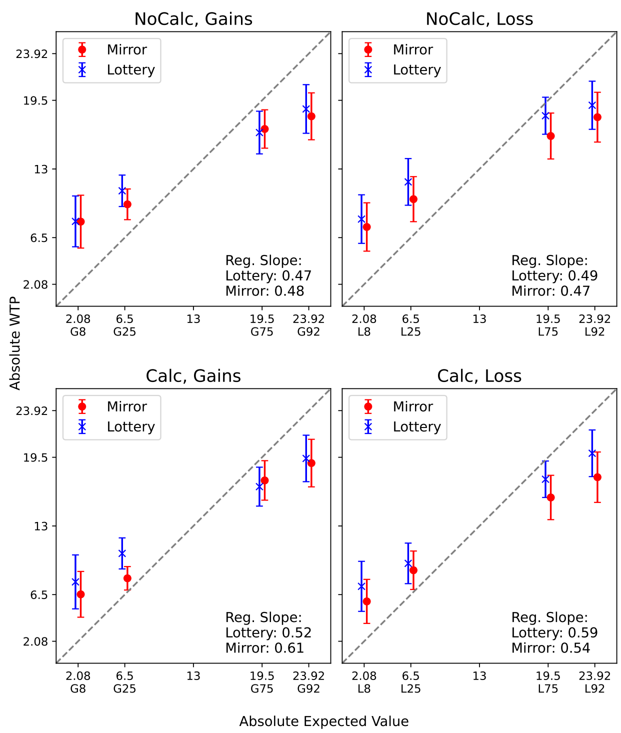


Figure D.5: Average valuation of each lottery and their deterministic mirror, only including the 49 subjects who answer all four batches of comprehension questions correctly at their first trial.

Appendix Figure D.6 replicates Figure 4. The two salient observations in the main text regarding the discriminability and the dispersion of valuations conditional on employing a group of base terms still hold.

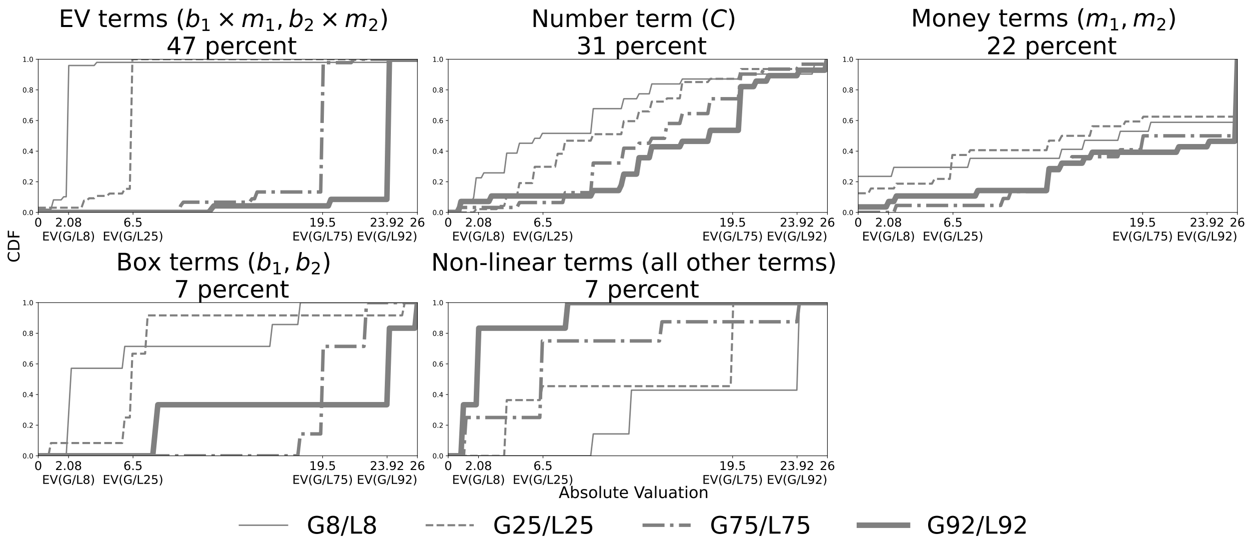


Figure D.6: Distributions of lottery valuations in the Calc treatment, conditional on employing each group of base terms, only including the 49 subjects who answer all four batches of comprehension questions correctly at their first trial.

Appendix Figure D.7 replicates Figure 5. The fraction of EV-types in the restricted sample is higher than that in the full sample. The EV-type subjects are still close to complete responsiveness (average responsiveness = 0.86). However, the non-EV-type subjects (average responsiveness = 0.33) are more responsive than in the full sample (0.25). Specifically, there are fewer non-EV-type subjects who exhibit extreme unresponsiveness in their lottery valuations. As a result, the bimodal distribution of responsiveness in the full sample has been replaced by the unimodal distribution in the restricted sample.

Appendix Figure D.8 replicates Figure 6. Again, the EV-types are more responsive than the non-EV-types as a whole in the NoCalc treatment, though the difference is a little bit smaller than when using the full sample, and the non-linear-types are a bit more responsive than the EV-types in the NoCalc treatment. Comparing between treatments by comparing Appendix Figure D.6 and Appendix Figure D.7, the EV-types again show significantly higher responsiveness in the Calc treatment.

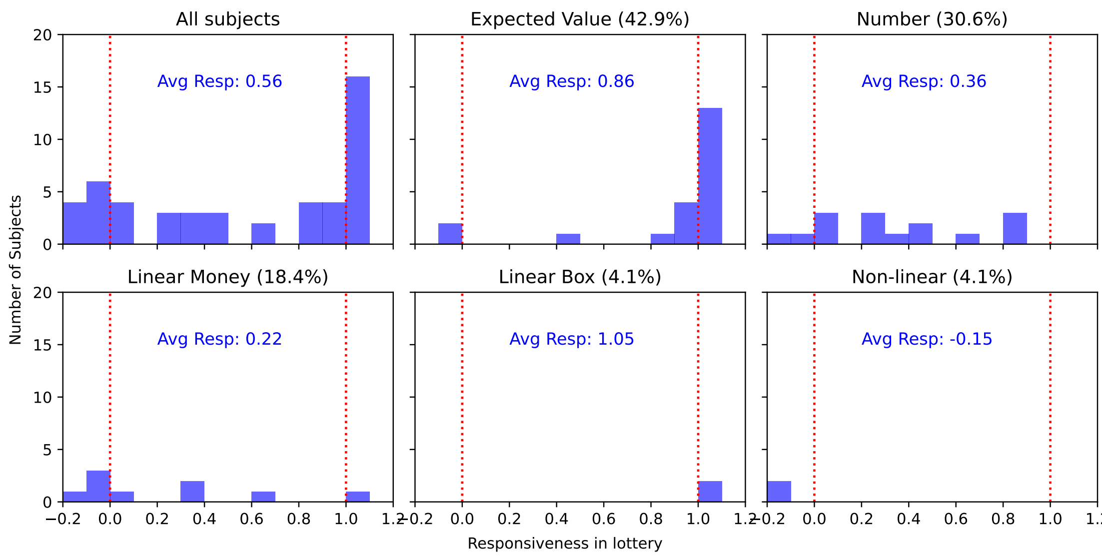


Figure D.7: Histograms of individual responsiveness in lottery tasks in the Calc treatment, only including the 49 subjects who answer all four batches of comprehension questions correctly at their first trial.

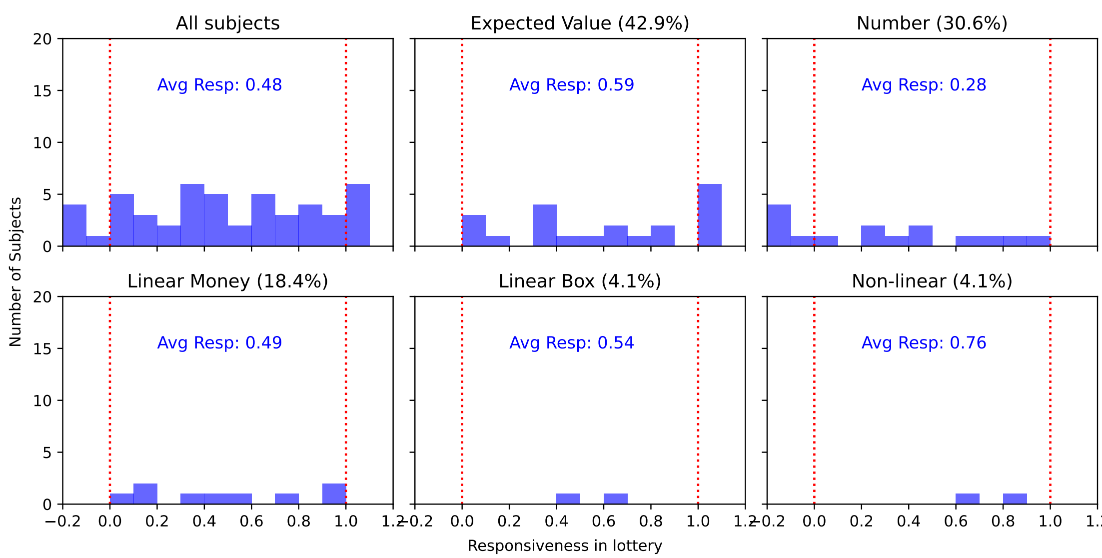


Figure D.8: Histograms of individual responsiveness in lottery tasks in the NoCalc treatment, only including the 49 subjects who answer all four batches of comprehension questions correctly at their first trial.

Does the weakening of the bimodality of responsiveness distribution invalidate the decomposition exercise in Section 6.1, where I show that the majority of the aggregate unresponsiveness should be attributed to the very responsive and the extremely unresponsive subjects, but not the moderately responsive subjects? The short answer is no. I replicate the exercise decomposing the aggregate unresponsiveness to the contributions of each subset of subjects in Section 6.1, and found that very responsive subjects ($r > 0.9$ in the Calc treatment) contribute to 25.0% of the aggregate unresponsiveness in the NoCalc treatment, the extremely unresponsive ($r < 0.1$ in the Calc treatment) contribute to 42.1%, and the rest 32.9%. The numbers are broadly comparable to those in Section 6.1, and the contribution of the moderately responsive subjects is only slightly higher than in the full sample.

E Reconciliation and Survey

E.1 Reconciliation

After the Calc treatment, all subjects enter the reconciliation stage, whose design follows Nielsen and Rehbeck (2022). Specifically, the subject is presented with a subset of all inconsistent valuations they submitted, and is given a chance to revise either of the choices to make them consistent, or leave the choices as they are. Inconsistent valuations are when a subject submits different valuations for the same task in the NoCalc and Calc treatment. For example, a subject may submit a valuation of \$10 for the task (G25, Lottery) in the NoCalc treatment, but instead \$6.5 for the same task (G25, Lottery) in the Calc treatment.

In the reconciliation stage, a subject is presented with the task to which they submit inconsistent valuations, the valuations they submit in both treatment, and the calculator input recorded in the Calc treatment. I explain to the subject that their valuations are inconsistent for this task, and the subject is then given a chance to reconcile their inconsistent valuations by choosing among three options: (1) Selecting the Calc valuation (changing their valuation in the NoCalc treatment to that in the Calc treatment); (2) Selecting the NoCalc valuation (changing their valuation in the Calc treatment to that in the NoCalc treatment); (3) Keep the valuations inconsistent as they are. The subject is not allowed to change both the valuations in NoCalc and Calc. The order of the three options is randomized between-subject, but fixed within-subject. The instructions and the subject interface can be found in Appendix G.[23]

The reconciliation stage is incentivized – if the task is randomly chosen to determine the subject's payment, the outcome of the BDM mechanism will depend on the valuation after reconciliation.

[23]Each subject faces at most four reconciliation tasks. If a subject submits three or more inconsistent valuations in lottery tasks, they will reconcile three randomly chosen inconsistent lottery valuations. Otherwise, they will reconcile all their inconsistent lottery valuations. The rest of the reconciliation tasks involve inconsistent mirror valuations.

type	Calc	NoCalc	Keep Inconsistent
expected value	55%	17%	28%
number	38%	29%	33%
linear money	44%	20%	35%
linear box	27%	30%	43%
non-linear	57%	19%	24%

Table E.1: Percentages of inconsistent valuations revised and direction of reconciliation for lottery tasks

Appendix Table E.1 shows the choices of whether to reconcile and the direction of reconciliation, by subject type (defined in Section 5.2) and for lottery tasks. The EV-type subjects select their Calc valuations in 55% of reconciliation tasks, while they only select their NoCalc valuations in 17% of tasks. I interpret this pattern as showing the responsive and near risk-neutral valuations submitted by the EV-type subjects in the Calc treatment better reflect their genuine risk attitudes than the moderately unresponsive and prospect theoretic valuations they submit in the NoCalc treatment. Appendix Figure E.1 further corroborates this interpretation by showing the CDF of the selected and unselected valuations by lottery task for the EV-type subjects, conditional on a reconciliation (i.e., not keeping valuations inconsistent as they are). The selected valuations have larger mass around the expected values than the unselected valuations.

As for the non-EV types, the number-type subjects show a much smaller gap between the two different directions of reconciliation. This should be expected since many number-type subjects do not perform any calculation in the Calc treatment and directly submit a number, which makes Calc similar to NoCalc. The linear money and non-linear types select their Calc valuations much more often than their NoCalc valuations, while the linear box type select their Calc and NoCalc valuations at similar frequencies.

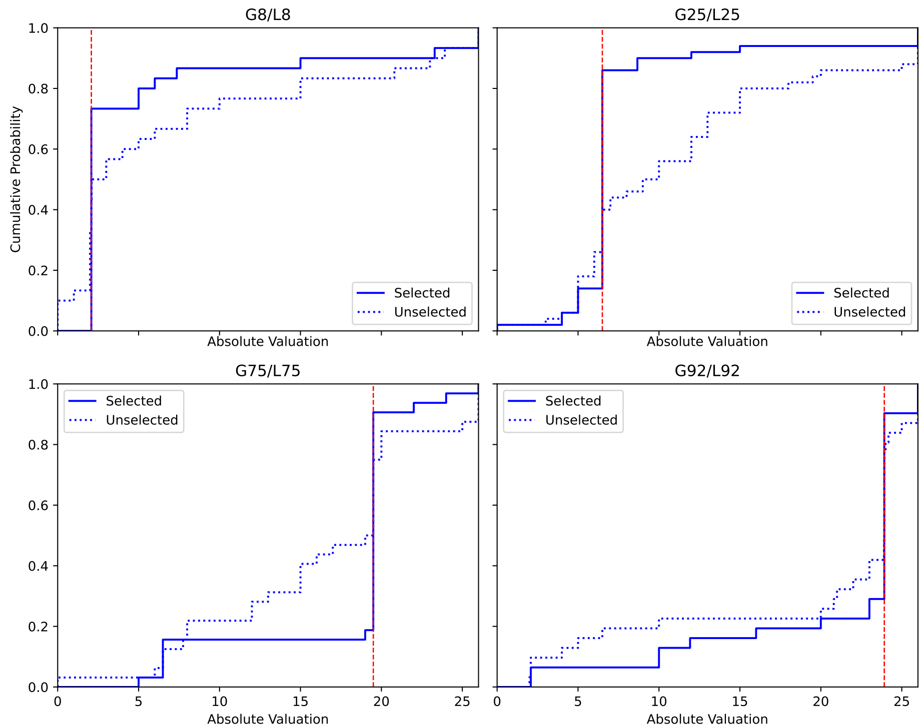


Figure E.1: CDF of selected and unselected valuations for EV-type subjects. The expected value of each lottery is marked with the red vertical line.

E.2 Survey

After the main tasks and the reconciliation stage, I included two additional questions testing subjects' basic understanding of lotteries. No calculator is provided for either question. In the first question (referred to as *EV*), I show subjects 100 boxes, 30 of which contains \$20, and the remaining \$0. The question asks subjects for the average amount of money in these 100 boxes. With elementary knowledge of arithmetic and some attentiveness, a subject can easily tell that the answer should be \$6 without the need to incur much implementation costs. The EV question is unincentivized. The second question (referred to as *FOSD*) is a choice task where the subject needs to choose among four mirrors. The mirrors are designed such that one of them apparently dominates the other three by having both (weakly) higher positive money amounts, and more boxes containing a positive amount of money. For a subject who understands the basics of the payment rule, the choice should be obvious after comparing the primitives of the sets of boxes and without the need to perform any calculations. The FOSD question is incentivized, and the mirror a subject chooses will be paid out with certainty.

The specific questions can be found in Appendix G.

type	%EV	%FOSD
expected value	96.1%	93.5%
linear box	60.0%	90.0%
linear money	58.8%	88.2%
nonlinear	42.9%	100.0%
number	71.6%	81.1%

Table E.2: Fraction of subjects who correctly answer each question

Appendix Table E.2 shows the fraction of subjects who correctly answer each question, by subject type. The vast majority of EV-type subjects correctly answer the EV question, while many non-EV-type subjects make mistakes. Appendix Table E.3 shows the average responsiveness for subjects who answer the EV question correctly and incorrectly, respectively, conditional on type. Generally speaking, responsiveness is higher in subjects who correctly answers the EV question, even conditional on type, except for the nonlinear types. In the meantime, for most non-EV types (except for linear box), the responsiveness is still quite low even conditional on correctly answering the EV question.

type	%(ev=6)	resp|ev=6	resp|ev $\neq$ 6
expected value	0.96	0.85	0.86
number	0.72	0.30	0.13
linear money	0.59	0.28	0.08
linear box	0.60	0.84	0.24
nonlinear	0.43	-0.20	0.04

Table E.3: Average responsiveness in lottery tasks in the Calc treatment, conditional on type and their correctness in the EV question

From Appendix Table E.2, it is also clear that the vast majority of subjects answer the FOSD question correctly. Although the FOSD question is incentivized and thus cannot be directly compared with the unincentivized EV question.

F Mathematical Expressions and Algorithms

This section first introduces a tree structure of mathematical expressions. Then, it explains the algorithm with which I recover the symbolic expressions, and the algorithm constructing the base terms. I implement the tree structure and the algorithm using the open-source package `sympy` in Python.

F.1 Mathematical Expressions as Trees

A mathematical expression can be represented as an *expression tree*. The tree has a few features:

- Leaf nodes represent operands (numbers or symbols).
- Non-leaf nodes represent operators (such as $+$, $\times$, or Power).
- Each non-leaf node generates a subtree, representing a sub-expression.
- Both $+$ and $\times$ are defined as n-ary operators (as opposed to binary operators), meaning they can take any finite number of arguments.
- The operators $-$ and $/$ are represented in the tree using $+$, $\times$, and Power. For example, $a - b$ is represented as $a + (-1 \times b)$, and a/b is represented as $a \times \text{Power}(b, -1)$.
- The parent-child structure of the tree represents the order of operators. A parent operator is evaluated later than its children.

As an example, Appendix Figure F.1 shows the tree representation of the expression $5 \times 4 + 3 - 1$. This representation preserves the order of operations because $+$ is a parent node of the $\times$. For more information, I refer the Reader to the sympy documentation.

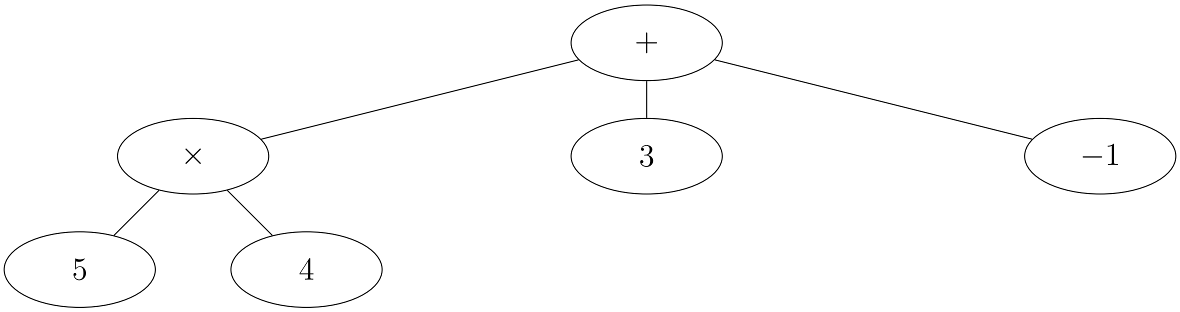


Figure F.1: The expression tree representing $5 \times 4 + 3 - 1$

For my purpose, one of the most important features of sympy's implementation of this expression tree is that any subtree with root operator $+$ or $\times$ will automatically "flatten" itself to the shallowest subtree possible. For example, an alternative (and illegal) expression tree representing the expression $5 \times 4 + 3 - 1$ is shown in Appendix Figure F.2. Since the operator $+$ is allowed to be n-ary (as opposed to binary), the right subtree will be automatically flattened to form the shallower tree in Appendix Figure F.1. For more information, see here for $+$ and here for $\times$.

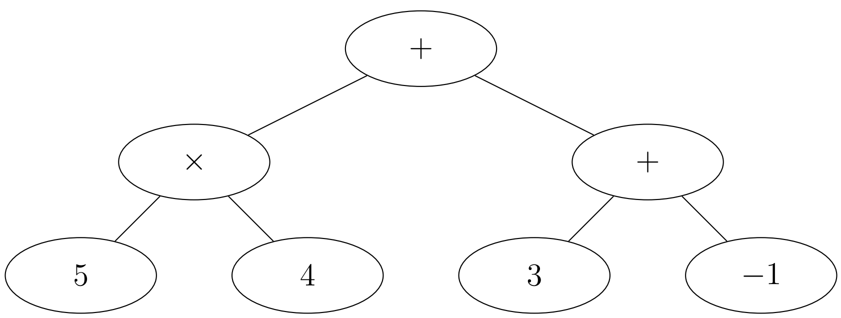


Figure F.2: An unflattened expression tree representing $5 \times 4 + 3 - 1$

F.2 Recovering Symbolic Expressions

I denote each calculator input as its sequence of calculator lines $L = (l_n)_n$, where l_n represents the numerical expression (represented as an expression tree) in line $n \in \{1, \ldots, \bar{n}\}$. I also use r_n to denote the numerical result from evaluating the numerical expression l_n. Let $P = \{(x_n, v_n)\}_n$ denote the set of task primitives that the algorithm matches against, where x_n represents a symbol or a symbolic sub-expression, and v_n represents its corresponding numerical value.

When implementing the algorithm, the following symbolic primitives are being matched:

$$\{b_1, b_2, m_1, m_2, b_1/100, b_2/100, m_1/100, b_1 \times m_1, b_1 \times m_1/100\}.$$

The set of primitives is expanded beyond $\{b_1, b_2, m_1, m_2\}$ to include common calculation shortcuts that subjects may use. For example, in G75, the primitive set includes $(b_1/100, 0.75)$ to capture the possibility that a subject divides the number of boxes by 100 implicitly in their mind before they use the result of this mental calculation directly in the calculator.

Having defined the necessary notations, I now describe the matching algorithm that recovers the symbolic expression from numerical expressions. Starting with the first line l_1, the algorithm iterates through its all leaf nodes (the numbers in the expression). If a number in a leaf node matches some v_n, the node's content is replaced with the corresponding symbolic sub-expression x_n. Any leaf node with a number that does not match any primitive remains as the same node. In this way, the algorithm constructs the symbolic expression for the first line, s_1.

Since the calculations made in a line $n > 1$ may build upon previous results, the algorithm must iteratively incorporate these intermediate calculations. Therefore, for each subsequent line $n > 1$, the algorithm expands its matching set to $M_n = (\cup_{i<n}(s_i, r_i)) \cup P$, which includes both the original primitives and all previous line results. Specifically, $\cup_{i<n}(s_i, r_i)$ contains the pairs of symbolic expressions and their computed results from all previous lines $i < n$. When processing line n, for any leaf node with a number that matches a previous result r_i, its content is replaced with the corresponding symbolic expression s_i. In the meantime, matches with primitive values v_k continue to be replaced with x_k. This process yields the symbolic expression s_n for each line n.

Through this iterative construction, the algorithm generates a sequence of symbolic expressions $S = (s_n)_n$ from the numerical expressions L.

F.3 Terms and Base Terms

For any symbolic expression l, if its root node is the operator $+$, its *terms* are all the second-level subtrees (whose roots are the immediate child nodes of the root node) of the symbolic expression. Otherwise, the symbolic expression has only one term: the symbolic expression itself. The flattening property of the expression trees ensures that all base terms have roots (the upper-level operand) other than the operator $+$. For example, $b_1 + b_2$ cannot be a term, while b_1 and $b_1 \times m_1$ are permitted. I implement the algorithm backing out terms via the function `as_ordered_terms` (see here) in sympy.

Then, for each term t, I first find all its factors. If the root node of a term is $\times$, the factors are all the second-level subtrees. Otherwise, the only factor of this term is the term itself. See the documentation of `as_ordered_factors` for more information. Finally, I drop all factors which are a single number (as opposed to symbols, or sub-expressions) from the term to generate its corresponding *base term*. If all factors are number factors (for example: (1) 4; (2) 4×2, as opposed to symbols or sub-expressions, the base term is defined as C.

The concept of terms, and by extension base terms, runs into an indeterminacy problem with syntactically different but mathematically equivalent expressions – for example, the mathematically equivalent expressions $b_1 \times m_1/100 + b_2 \times m_2/100$ and $(b_1 \times m_1 + b_2 \times m_2)/100$ lead to different terms and in turn base terms. To address this problem, I first expand all the products in all expressions by applying the distributive law of multiplication ($a \times (b + c) = a \times b + a \times c$), wherever applicable. This way, I transform the original symbolic expression into its distributed form expression. Using distributed form expressions solves the aforementioned indeterminacy problem and generates the same set of base terms ($\{b_1 \times m_1, b_2 \times m_2\}$) for $b_1 \times m_1/100 + b_2 \times m_2/100$ and $(b_1 \times m_1 + b_2 \times m_2)/100$.

G Experimental Instructions of the Main Experiment

General Information

Welcome to our study on decision-making.

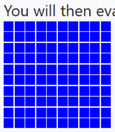
Participation in this study guarantees a $7.00 show-up fee. Additionally, you have the opportunity to earn a bonus. How much bonus you earn will depend partly on the decisions you make and partly on chance.

This session includes three Parts. You will be asked to make decisions in each Part, and the first two Parts offer you an opportunity to earn a bonus. The total bonus paid to you at the end of this session will be the sum of the bonuses you earn in all Parts.

Next

Boxes with Money

- In each of several tasks, we will give you an **INITIAL** sum of money of **$30**.
- You will then evaluate a set of **100 BOXES** which the computer may open to **either increase or decrease** this initial sum.



- Each box contains either a **POSITIVE** or **NEGATIVE** amount of money (or nothing). When the computer opens one or more boxes from a set, the amount of money in the opened boxes will be added to (or subtracted from) your INITIAL money to determine your **BONUS**.

Next

The Decision Table

- Each set of boxes will be described in a **TABLE** like the one below. For each set, one or more counts of boxes (for instance 75, 25 or 100 boxes) are listed at the top, and the positive or negative amount of money in that number of boxes (for instance $20, $0, $7) is shown in the row of the Table.
-

75 Boxes	25 Boxes
$20.00	$0.00

In the example above, the set consists of 75 boxes each containing $20 and 25 boxes each containing $0.

- In the example below, the set consists of 25 boxes with -$12 (negative $12) in each box and 75 boxes with $0 in each box.

25 Boxes	75 Boxes
- $12.00	$0.00

- Depending on the task, your job will be to decide how much you'd be willing to pay to either **cause** the computer to open boxes from the set to modify your **BONUS** or **prevent** the computer from opening the boxes to modify your bonus.

Next

If the next tasks are lottery tasks, the following instructions appear.

A Random Box

- In the upcoming tasks, if the computer opens boxes, it will pay you by **RANDOMLY** selecting one of the 100 boxes (each box in the set is **EQUALLY** likely to be selected by the computer). If the amount in the box is positive, it will be **ADDED** to your initial money. If the amount is negative, it will be **SUBTRACTED** from your initial money.

- Example: In the example below, there are 100 boxes in the set. For this set, 50 boxes contain $16.00 and 50 of them contain $0.00. If the computer opens the boxes, there is therefore a 50% chance that $16 will be added to your initial amount of money and a 50% chance $0 will be added.

50 Boxes	50 Boxes
$16.00	$0.00

- Example: In the example below, there are also 100 boxes in the set. For this set, 50 boxes contain -$8.00 and 50 of them contain $0.00. If the computer opens the boxes, there is therefore a 50% chance you will have $8 subtracted from your initial amount of money (you lose $8) and a 50% chance that you have $0 subtracted.

50 Boxes	50 Boxes
- $8.00	$0.00

- Depending on the task, your job will be to decide how much you'd be willing to pay to either cause the computer to open boxes from the set to modify your **BONUS** or prevent the computer from opening the boxes to modify your bonus.

Instead, if the next tasks are mirror tasks, the following instructions appear.

The Average Box

- In the upcoming tasks, if the computer opens boxes, it will pay you by calculating the **AVERAGE** amount of money across all 100 boxes. That is, it will add up the amount of money from each of the 100 boxes and divide that sum by 100. If the amount is positive, that amount will be **ADDED** to your initial money. If the amount is negative, it will be **SUBTRACTED** from your initial money.

- Example: In the example below, there are 100 boxes in the set. For this set, 50 boxes contain $16.00 and 50 of them contain $0.00. If the computer opens these boxes, it will therefore add $8 to your initial amount of money for certain, which is the **AVERAGE** amount of money across the 100 boxes.

50 Boxes	50 Boxes
$16.00	$0.00

- Example: In the example below, there are also 100 boxes in the set. In this set, 50 boxes contain -$8.00 and 50 of them contain $0.00. If the computer opens these boxes, the computer will pay you -$4 for your choice, which is the **AVERAGE** amount of money across the 100 boxes; it will therefore subtract $4 from your initial amount of money (that is, you will lose $4) for certain.

50 Boxes	50 Boxes
- $8.00	$0.00

Four comprehension questions are asked on the same page as the instructions over the payoff rules of lotteries. The subject needs to simultaneously answer all four questions correctly to proceed. The answers to the first three questions differ across lottery and mirror.

Please answer the questions below.

50 Boxes	50 Boxes
$16.00	$0.00

Suppose that this example of boxes determines your payment in the upcoming tasks.

For the upcoming tasks, how will your payment be decided by the boxes?

- The maximum box
- The minimum box
- The average box
- A random box

What is the chance that $16 is added to your earnings?

- 0 in 100 (0%)
- 50 in 100 (50%)
- 100 in 100 (100%)

What is the chance that $8 is added to your earnings?

- 0 in 100 (0%)
- 50 in 100 (50%)
- 100 in 100 (100%)

What is the chance that $4 is added to your earnings?

- 0 in 100 (0%)
- 50 in 100 (50%)
- 100 in 100 (100%)

Next

After introducing the payout rules of lottery (or mirror), instructions for the BDM mechanism is shown. First, all subjects are shown the instructions for the gain tasks, eliciting the willingness to pay.

Paying for a Set of Boxes

- In the experiment, we will ask you the maximum amount you would be willing to pay either to cause or prevent the computer from opening the set of boxes on your screen to modify your Initial earnings.
- In some tasks (colored in green) we will show you a set that contains positive amounts of money, and ask you to tell us how many dollars you would (**at the very maximum**) be willing to pay to **cause** the computer to open boxes from the set to increase your earnings. Or, in other words, we will ask you how much do you think it is worth to you to have these boxes influence your earnings?
- Example: On your screen, we will show you a text box like the one below. Just enter the amount of money you think the set is worth to you (the maximum amount you'd be willing to pay for the set to be opened - the screen will give you the range you can enter):

  I would be willing to pay **a maximum of**:

  (enter a number between $0 and $25.00)
- To **reward you** for giving an **honest answer**, we are going to use a special set of rules to determine your payments in these tasks. We will randomly pick a **price** (equally likely between 0 and the maximum value you are allowed to enter) for the set of boxes (you won't know the price when you make your choice). If the amount you entered is **greater than or equal to** that random price, the computer will open the set of boxes on the screen to modify your Initial earnings as described **and** you will pay the amount of the random price (not the amount you entered) from your total earnings. If your maximum amount is less than the random price, the computer will not open the boxes on your screen and you will simply earn your initial amount (and you will not pay the random price).
- Important: If the computer uses boxes from the set to modify your earnings, you will not have to pay the maximum amount you enter, but instead will pay the random price. The maximum amount you enter just lets you tell us the range of random prices you are willing to pay for the set of boxes.
- If this sounds confusing, it is **actually very simple**. We've designed the payments so it is in your best interest to **tell us honestly** the **most** you would be willing to pay to have the set of boxes opened to influence your bonus. So just think about how much at a maximum you'd be willing to give up to have the computer modify your bonus based on the set of boxes on your screen, and enter this amount truthfully.
- Example: In the example below, the set consists of $12 in each of the 100 boxes. If the computer opens the boxes, with certainty $12 will be added to your initial amount of money.

100 Boxes	0 Boxes
$12.00	$0.00

  To maximize the bonus you get from this session, you should submit $12 as the maximum amount that you are willing to pay to have the computer open the boxes. Our payment rule guarantees this.

Then, all subjects are shown the instructions for the gain tasks, eliciting the willingness to prevent (negative of the valuation).

Paying to Avoid a Set of Boxes

- In other tasks (colored in red) we will show you a set that contains negative amounts of money, and ask you to tell us how many dollars you would (at the very maximum) be willing to pay to prevent the computer from opening boxes from the set to decrease your earnings. Or, in other words, we will ask you how much do you think it is worth to you to prevent these boxes from influencing your earnings?

- Example: On your screen, we will, again, show you a text box like the one below. Just enter the amount of money you think it is worth to prevent the computer from using that set to modify your bonus (the maximum amount you'd be willing to pay to prevent it - the screen will give you the range you can enter):

  I would be willing to pay **a maximum of**:

  (enter a number between $0 and $25.00)

- To **reward you** for giving an **honest answer**, we are going to use a special set of rules to determine your payments in these tasks. We will randomly pick a **price** (equally likely between 0 and the maximum value you are allowed to enter) for the set of boxes (you won't know the price when you make your choice). If the amount you entered is **greater than or equal to** that random price, the computer will **not open** the set of boxes on the screen to modify your Initial earnings as described **and** you will pay the amount of the random price (not the amount you entered) from your total earnings. If your maximum amount is less than the random price, the computer will open the boxes on your screen and you will simply earn your initial amount (and you will not pay the random price).

- Important: If you prevent the computer from opening boxes from the set, you will not have to pay the maximum amount you enter, but instead will pay the random price. The maximum amount you enter just lets you tell us the range of random prices you are willing to pay to avoid the set of boxes.

- If this sounds confusing, it is **actually very simple**. We've designed the payments so it is in your best interest to **tell us honestly** the **most** you would be willing to pay to prevent the set of boxes from being opened to influence your bonus. So just think about how much at a maximum you'd be willing to give up to prevent the computer from modifying your bonus based on the set of boxes on your screen, and enter this amount truthfully.

- Example: In the example below, the set consists of -$12 in each of the 100 boxes. If the computer opens the boxes, with certainty $12 will be subtracted from your initial amount of money.

100 Boxes	0 Boxes
- $12.00	$0.00

To maximize the bonus you get from this session, you should submit $12 as the maximum amount that you are willing to pay to prevent the computer from opening the boxes. Our payment rule guarantees this.

The last part of instructions before the main tasks.

Several Sets of Boxes

- Over the course of the session, we will show you several sets of boxes. Each gives you $30 **initial amount of money**, but may have **different** amounts of money distributed across the boxes.
- Important: Make sure you pay attention to the type of question we are asking in each task. In some tasks colored in green we are asking you to tell us how much you'd be willing to pay to **cause** the boxes to influence your earnings. In other tasks colored in red we are asking you to tell us how much you'd be willing to pay to **prevent** the boxes from influencing your earnings.
- **One out of five (1/5 of)** participants will be randomly selected by the computer to be paid a **BONUS** based on their choices. If you are one of these participants, at the end of the session the computer will **RANDOMLY** select **ONE** Task and then **RANDOMLY** select a **PRICE** to determine your payment based on how much you said you're willing to pay.
- Since you do not know which choice will be selected, you should make each choice as if it alone determines your payment.

Experimental interface in the NoCalc treatment:

Initial money: $30.00

75 Boxes	25 Boxes
$26.00	$0.00

I would be willing to pay **a maximum of**:

(enter a number between $0 and $26.00)

to **have** a **randomly selected** box's contents **added to** my Initial Money.

Remember, we've designed the payments so it is in your best interest to **tell us honestly** the most you would be willing to pay to have the set of boxes opened to influence your bonus. So just think about **how much at a maximum you'd be willing to give up** to have the computer modify your bonus based on the set of boxes on your screen, and enter this amount truthfully.

Next

After completing the first block of 9 tasks with the first payment rule, the subjects are notified with a change of payment rule.

The Average Box

- Starting from now, we will change the way in which you will get paid.
- In the upcoming tasks, if the computer opens boxes, it will pay you by calculating the **AVERAGE** amount of money across all 100 boxes. That is, it will add up the amount of money from each of the 100 boxes and divide that sum by 100. If the amount is positive, that amount will be **ADDED** to your initial money. If the amount is negative, it will be **SUBTRACTED** from your initial money.
- Example: In the example below, there are 100 boxes in the set. For this set, 50 boxes contain $16.00 and 50 of them contain $0.00. If the computer opens these boxes, it will therefore add $8 to your initial amount of money for certain, which is the **AVERAGE** amount of money across the 100 boxes.

50 Boxes	50 Boxes
$16.00	$0.00

- Example: In the example below, there are also 100 boxes in the set. In this set, 50 boxes contain -$8.00 and 50 of them contain $0.00. If the computer opens these boxes, the computer will pay you -$4 for your choice, which is the **AVERAGE** amount of money across the 100 boxes; it will therefore subtract $4 from your initial amount of money (that is, you will lose $4) for certain.

50 Boxes	50 Boxes
- $8.00	$0.00

A Random Box

- Starting from now, we will change the way in which you will get paid.
- In the upcoming tasks, if the computer opens boxes, it will pay you by **RANDOMLY** selecting one of the 100 boxes (each box in the set is **EQUALLY** likely to be selected by the computer). If the amount in the box is positive, it will be **ADDED** to your initial money. If the amount is negative, it will be **SUBTRACTED** from your initial money.
- Example: In the example below, there are 100 boxes in the set. For this set, 50 boxes contain $16.00 and 50 of them contain $0.00. If the computer opens the boxes, there is therefore a 50% chance that $16 will be added to your initial amount of money and a 50% chance $0 will be added.

50 Boxes	50 Boxes
$16.00	$0.00

- Example: In the example below, there are also 100 boxes in the set. For this set, 50 boxes contain -$8.00 and 50 of them contain $0.00. If the computer opens the boxes, there is therefore a 50% chance you will have $8 subtracted from your initial amount of money (you lose $8) and a 50% chance that you have $0 subtracted.

50 Boxes	50 Boxes
- $8.00	$0.00

- Depending on the task, your job will be to decide how much you'd be willing to pay to either cause the computer to open boxes from the set to modify your **BONUS** or prevent the computer from opening the boxes to modify your bonus.

The same set of 4 comprehension questions are asked again.

After the NoCalc treatment, the subjects enter the Calc treatment and are shown the next instructions.

Calculator Provided

You may have found that in doing the task you want to perform some calculation to the numbers. Accompanying all upcoming tasks, a calculator will be provided to you on your screen to help you.

Its up to you how you use the calculator. But **you need to submit the maximum amount that you are willing to pay using the calculator.** If you are chosen to be paid by the computer, you will be paid only according to the maximum amount you submit, not what you calculate.

How you submit your response in the calculator

For each task, we record your maximum amount willing to pay by taking the number in the *Result* column of the most recent line of your calculator as your response.

We now show you how to submit your response with the calculator using an example. **We deliberately choose an example unrelated to our task to avoid prejudicing you towards a particular way of valuing the boxes**.

Example: Imagine someone is asked to convert 32 kilograms to ounces, and submit their result in the calculator. They may do the following:

- First, they calculate $32 \times 2.205 = 70.56$ to convert 32 kilograms to 70.56 pounds (1 kilogram = 2.205 pounds)
- Second, they calculate $70.56 \times 16 = 1128.96$ to convert 70.56 pounds to 1128.96 ounces (1 pound = 16 ounces)
- Finally, they submit 1128.96 as their response.

We now show you how to achieve this in the calculator.

As a first step, you perform the steps as described above in the calculator until the most recent line in the *Result* column (Line 2 in this example) shows your intended response.

As a first step, you perform the steps as described above in the calculator until the most recent line in the *Result* column (Line 2 in this example) shows your intended response.

	Calculation	Result
1	32×2.205	70.56
2	70.56×16	1128.96
3		

Then, you can click *Fill*, and the number in the most recent line in the *Result* column will be filled into the textbox asking for your response. The textbox is greyed out, so that you can only use the calculator to fill numbers into it.

I would be willing to pay **a maximum of**:

1128.96

You will be asked to double check whether the recorded response matches your intended response. If they do, click the "Submit" button, and then your response to this task will be recorded, and your will enter the next task. If they do not, you can modify the calculation steps until it leads to what you would like to submit.

	Calculation	Result
1	32×2.205	70.56
2	70.56×16	1128.96
3		

Ans	(	)	Del
7	8	9	/
4	5	6	×
1	2	3	−
0	.	Enter	+
Submit	You are about to submit **1128.96** as your response. Are you sure?		

For your task, you will similarly use the *Fill* and *Submit* buttons to submit the maximum amount that you are willing to pay to best advance your interests.

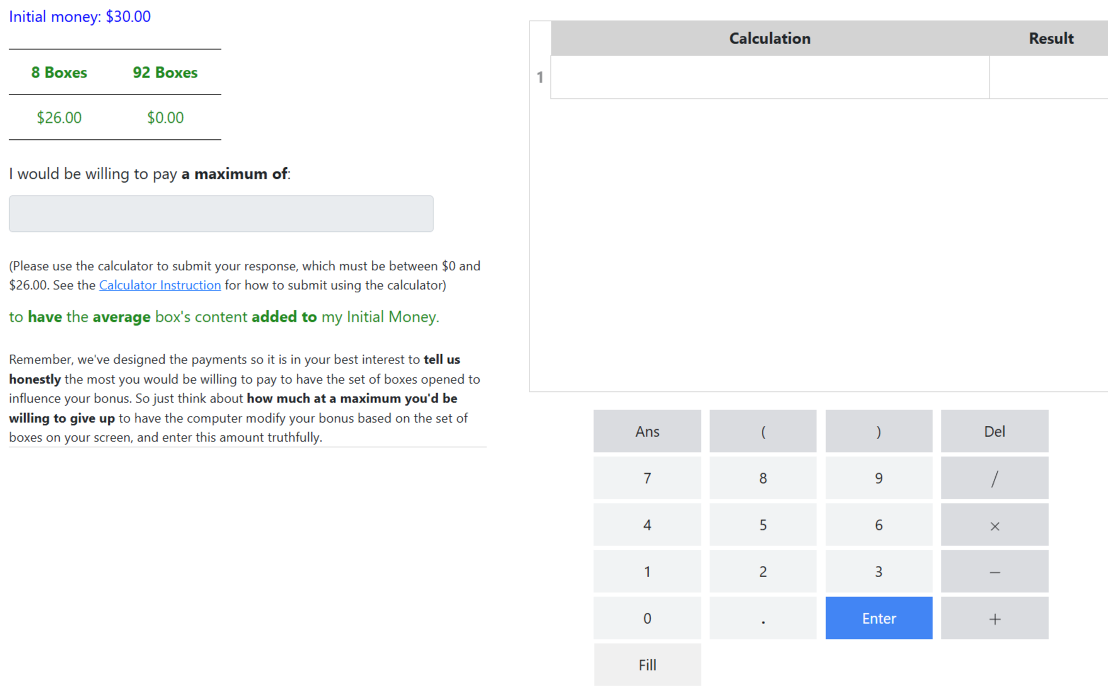
You may want to try the example above with the calculator provided below. **Also, you may want to try using the keyboard to type in the calculator.** For example, you can use `1` to type *1*, `*` to type ×, `Backspace` to type *Del*.

	Calculation	Result
1		

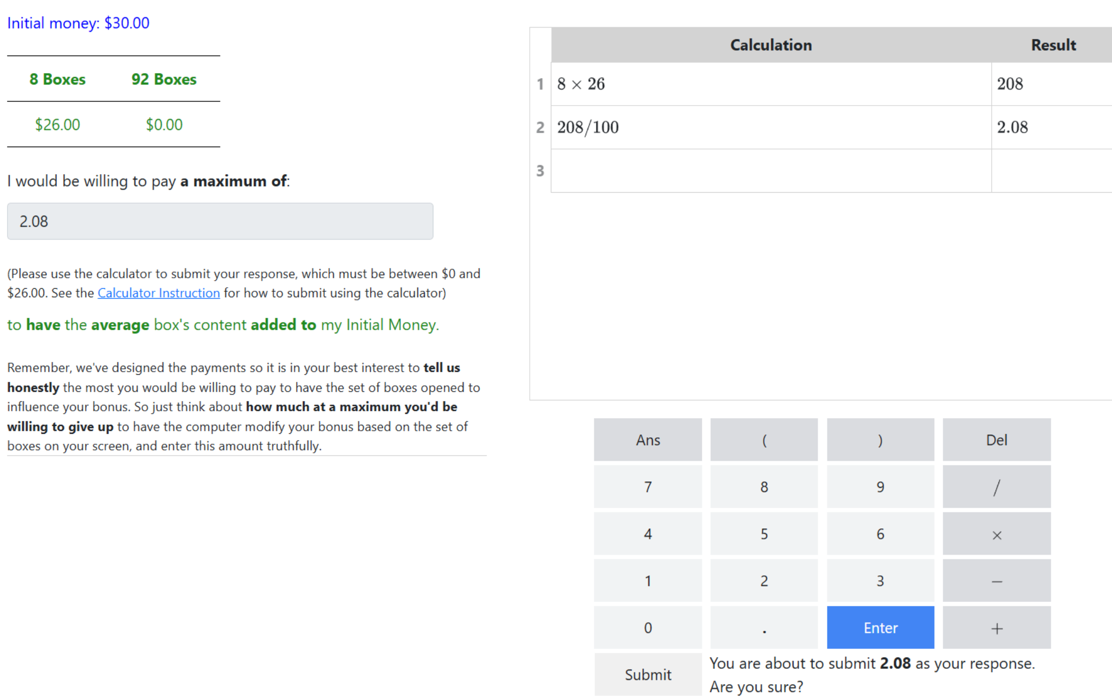
Ans	(	)	Del
7	8	9	/
4	5	6	×
1	2	3	−
0	.	Enter	+
Fill			

Next

Experimental interface in the Calc treatment, before the subject performs any calculation:



Experimental interface in the Calc treatment, after the subject performs some calculations and clicked *Fill*:



The interface in the reconciliation tasks (data analyzed in Appendix E).

Revising Your Choices

Below, we show another task where you responded inconsistently. We are giving you a chance to revise your response to the task.

The Inconsistent Task

For the **random box** task showed below:

Initial money: $30.00

92 Boxes	8 Boxes
$26.00	$0.00

I would be willing to pay **a maximum of**:

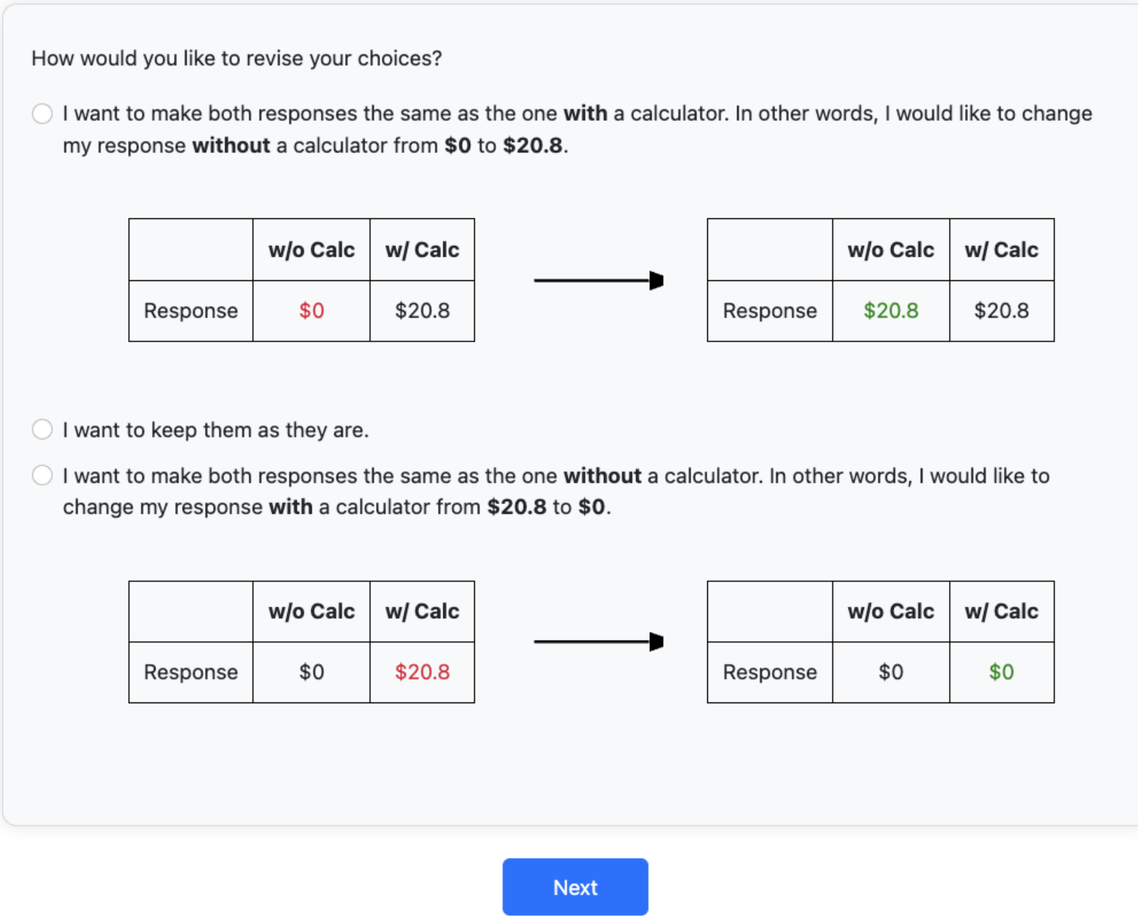
to **have** a **randomly selected** box's contents **added to** my Initial Money

You gave a maximum of **$0 without** a calculator.

However, you also gave a maximum of **$20.8 with** a calculator. Below shows what you did with the calculator.

	Calculation	Result
1	26×0.8	20.8

The order of the three options is randomized across subjects, but fixed for the same subjects across reconciliation tasks.



How would you like to revise your choices?

- I want to make both responses the same as the one **with** a calculator. In other words, I would like to change my response **without** a calculator from **$0** to **$20.8**.

	w/o Calc	w/ Calc
Response	$0	$20.8

	w/o Calc	w/ Calc
Response	$20.8	$20.8

- I want to keep them as they are.
- I want to make both responses the same as the one **without** a calculator. In other words, I would like to change my response **with** a calculator from **$20.8** to **$0**.

	w/o Calc	w/ Calc
Response	$0	$20.8

	w/o Calc	w/ Calc
Response	$0	$0

Next

After the reconciliation task, the subject is asked to perform a choice task between four mirrors (data analyzed in Appendix E).

A Choice Problem

In Part 2, you are asked to choose between a few Sets of Boxes.

With certainty, your choice of the Set of Boxes will be implemented and you will receive a bonus based on the **AVERAGE** box in the Set you choose. The bonus will be added to your total earnings at the end of the session.

It is worth reminding yourself of how the Average Box works: Under Average Box, the computer will pay you by calculating the **AVERAGE** amount of money across all 100 boxes in your chosen Set.

Box Set 1

19 Boxes	81 Boxes
$2.10	$0.00

Box Set 2

32 Boxes	68 Boxes
$4.20	$0.00

Box Set 3

32 Boxes	68 Boxes
$2.10	$0.00

Box Set 4

19 Boxes	81 Boxes
$4.20	$0.00

Which set of boxes do you prefer it to modify your bonus?

- Box Set 1
- Box Set 2
- Box Set 3
- Box Set 4

Next

Finally, the subject is asked a simple question testing their understanding of the concept of *averages* (data analyzed in Appendix E). The question is unincentivized, and the page title "Incentivized Survey" is a typo.

Incentivized Survey

Now, you are asked to answer a question as the last part of this session. You will not be paid for this question.

Imagine you are facing the set of boxes below

30 Boxes	70 Boxes
$20.00	$0.00

What is the **average** amount of money in these 100 boxes?

Next

H Experimental Instructions for the Replication Adopting Instructions from Wu (2025)

The instructions differ from those in Appendix G in four aspects: (1) How the lotteries and mirrors are described; (2) How to ensure subjects' comprehension of the different payment rules in lotteries and mirrors; (3) How to screen out subjects when they fail comprehension checks; and (4) How the BDM mechanism is described to subjects.

At the beginning, the subjects are introduced with the first payment rule. The screenshot below shows the description of the payment rules for lottery and mirror, respectively.

Boxes with Money

All choice tasks in this study will involve a set of boxes paying out money according to a payment rule.

Boxes paying out money

In each task of this study, you will be facing 100 boxes that pay out money according to a payment rule.

Example:

30 Boxes	70 Boxes
$100	$0

Payment rule: The computer will calculate the outcome by **randomly picking** one of the boxes. The amount of money in that box (in this example, could be $0 or $100) will be your final result.

Boxes with Money

All choice tasks in this study will involve a set of boxes paying out money according to a payment rule.

Boxes paying out money

In each task of this study, you will be facing 100 boxes that pay out money according to a payment rule.

Example:

30 Boxes	70 Boxes
$100	$0

Payment rule: You will receive the **average amount of money** in the set of boxes **with certainty**. In this example, you would receive $30, obtained by ($100*30+$0*70)/100 = $30.

The subjects are then shown 10 randomly simulated outcomes from the current payment rule, using 3 example sets of boxes which all involve an non-zero outcome of $100. One example set of boxes are shown in the screenshot, under the lottery payment rule.

Simulated Payment from the Boxes

Example:

75 Boxes	**25 Boxes**
$100	$0

To help you better understand the possible outcomes, we simulate 10 random outcomes from the boxes above.

- $100
- $100
- $100
- $0
- $0
- $100
- $0
- $100
- $100
- $100

Next, the subjects go through the comprehension check. A total of 5 questions are asked, and if the subject answers 4 or fewer correctly, they will be screened out of the experiment. The questions differ in the set of boxes shown, which all involve an non-zero outcome of $100.

Next Step: Comprehension Check

You'll be asked a few questions to make sure you understand the task. To pass the check, you need to answer 4 out of 5 questions correctly.

If you answer fewer than 4 questions correctly, you will be screened out of the study. You will not be paid for your participation in this case.

If you need to review the instructions, a link will be available for quick access.

Feel free to proceed when you're ready!

Next

Comprehension Question

Imagine you have the following option. If we simulate this option just like what we did, what possible outcomes might you see? Choose all that apply.

Review Instructions

50 boxes	50 boxes
$100	$0

☐ $0 ☐ $10 ☐ $30 ☐ $50 ☐ $70 ☐ $90 ☐ $100

Submit

The correct answer of the question above is 50 for mirror payment rule, and 0 and 100 for lottery payment rule.

Next, the subject gets into the descriptions of the BDM mechanism, adopted from Healy (2020) with minimal changes.

Your Task

Choosing Between Boxes and Money

- For each set of 100 boxes, we are going to ask you a list of questions like the following:

Q#	Question	Option A		Option B
1	Would you rather have:	**The Boxes**	*or*	**$0.01**
2	Would you rather have:	**The Boxes**	*or*	**$0.02**
3	Would you rather have:	**The Boxes**	*or*	**$0.03**
⋮	⋮	⋮	⋮	⋮
1,999	Would you rather have:	**The Boxes**	*or*	**$19.99**
2,000	Would you rather have:	**The Boxes**	*or*	**$20.00**

- In each question you pick either Option A (the boxes) or Option B (the money). After you answer all questions, we will randomly pick *one* question and pay you the option you chose on that one question. Each question is equally likely to be chosen for payment. Obviously you have no incentive to lie on any question, because if that question gets chosen for payment then you'd end up with the option you like less.

- To maximize the money you get, we assume you're going to choose the Option A in at least the first few questions, but at some point switch to choosing Option B. So, to save time, just tell us at which dollar value you'd switch. We can then 'fill out' your answers to all questions based on your switch point (choosing Option A for all questions before your switch point, and Option B for all questions at or after your switch point). We will still draw one question randomly for payment. Again, if you lie about your true switch point you might end up getting paid an option that you like less.

- To maximize the money you get, we assume you're going to choose the Option A in at least the first few questions, but at some point switch to choosing Option B. So, to save time, just tell us at which dollar value you'd switch. We can then 'fill out' your answers to all questions based on your switch point (choosing Option A for all questions before your switch point, and Option B for all questions at or after your switch point). We will still draw one question randomly for payment. Again, if you lie about your true switch point you might end up getting paid an option that you like less.

- We will ask you to tell us at which dollar value you'd switch.

- Example 1: In the example below, because the set of boxes consists of $12 in each of the 100 boxes, the boxes will pay out $12 with certainty.

100 Boxes	**0 Boxes**
$12.00	$0.00

To maximize the bonus you get, you should switch at $12. Why?

The subjects are also shown the instructions of the BDM mechanism for tasks involving losses.

Tasks Involving Losses

In this study, you may also encounter tasks that involve losses. In these tasks, you will be given an initial amount of money and asked to choose between a set of boxes that involve losses and a certain amount of loss.

Boxes involving losses

Example: You are given $100 as your initial amount of money.

30 Boxes	70 Boxes
-$100	$0

Payment rule: The computer will calculate the outcome by **randomly picking** one of the boxes. The amount of money in that box (in this example, could be $0 or -$100) will be deducted from your initial amount of money. The remaining amount of money will be paid to you.

Choosing Between Boxes and Money

- For these tasks, we are going to ask you a list of questions like the following:

Q#	Question	Option A		Option B
1	Would you rather have:	**The Boxes**	*or*	**Losing $0.01**
2	Would you rather have:	**The Boxes**	*or*	**Losing $0.02**
3	Would you rather have:	**The Boxes**	*or*	**Losing $0.03**
⋮	⋮	⋮	⋮	⋮
1,999	Would you rather have:	**The Boxes**	*or*	**Losing $19.99**
2,000	Would you rather have:	**The Boxes**	*or*	**Losing $20.00**

- To minimize your loss, we assume you're going to choose the Option B in at least the first few questions, but at some point switch to choosing Option A. So, to save time, just tell us at which dollar value of loss you'd switch. We can then 'fill out' your answers to all questions based on your switch point (choosing Option B for all questions before your switch point, and Option A for all questions at or after your switch point). We will still draw one question randomly for payment. Again, if you lie about your true switch point you might end up getting paid an option that you like less.
- We will ask you to tell us at which dollar value you'd switch.

- We will ask you to tell us at which dollar value you'd switch.

- Example 1: In the example below, because the set of boxes consists of -$12 in each of the 100 boxes, the boxes will involve a loss of $12 with certainty.

100 Boxes	0 Boxes
-$12.00	$0.00

To maximize the bonus you get, you should switch at $12. Why?

Then the subjects enter the NoCalc treatment with the interface below.

Click here to review instructions

8 Boxes	92 Boxes
-$26.00	$0.00

Q#	Option A	Option B
1	The Boxes	Losing $0.01
2	The Boxes	Losing $0.02
3	The Boxes	Losing $0.03
⋮	⋮	⋮
2,599	The Boxes	Losing $25.99
2,600	The Boxes	Losing $26.00

At which dollar value would you switch?

(enter a number between $0 and $26.00)

Next

After completing the first block of 9 tasks with the first payment rule, the subjects are notified with a change of payment rule.

Boxes with Money

In the upcoming choice tasks, the **payment rule has changed**. Please review the new payment rule below.

Boxes paying out money

In each task of this study, you will be facing 100 boxes that pay out money according to a payment rule.

Example:

30 Boxes	70 Boxes
$100	$0

Payment rule: You will receive the **average amount of money** in the set of boxes **with certainty**. In this example, you would receive $30, obtained by ($100*30+$0*70)/100 = $30.

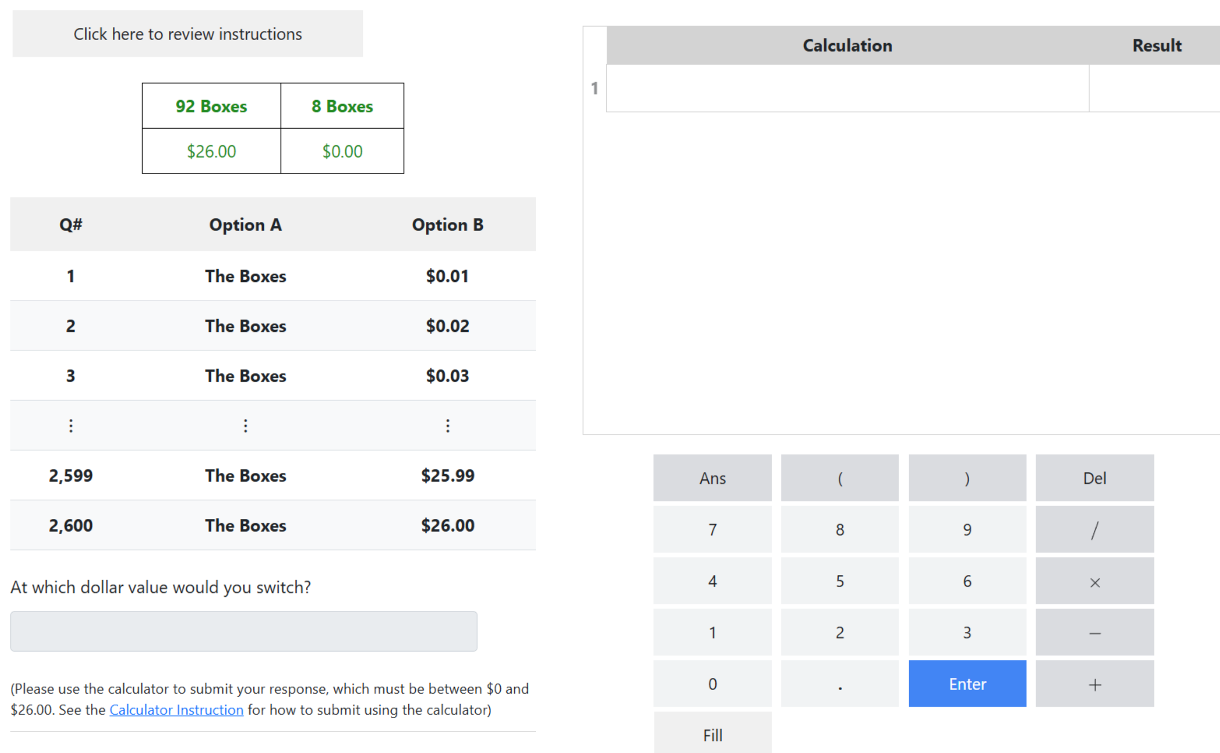
Next

Then, the subjects again go through the 3 simulated outcomes and the 5 comprehension questions. Again, if the subjects answer 4 or fewer comprehension questions correctly, they will be screened out of the experiment and will be paid $2.50 for their time.

If they pass the second set of comprehension questions, they will enter the second block of the NoCalc treatment.

After finishing the NoCalc treatment, the subjects enter the Calc treatment and are shown the calculator instructions shown in Appendix G.

Then, the subjects will again be notified the change of payment rule that goes back to their first block in the NoCalc treatment. The 3 simulated outcomes are shown again, but there is no comprehension questions and subject screening in the Calc treatment.



The remaining instructions are exactly the same as Appendix G, except for the correction of the typo in the last screenshot of Appendix G.